(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) **EP 3 920 102 B1**

(12) **EUROPEAN PATENT SPECIFICATION**

(45) Date of publication and mention of the grant of the patent:
30.10.2024 Bulletin 2024/44

(21) Application number: **21177789.1**

(22) Date of filing: **04.06.2021**

(51) International Patent Classification (IPC):
G06N 3/045 *(2023.01)* ***G06N 5/01*** *(2023.01)*
G06N 3/0985 *(2023.01)* ***G06N 3/09*** *(2023.01)*

(52) Cooperative Patent Classification (CPC):
G06N 3/045; G06N 3/09; G06N 3/0985; G06N 5/01

(54) **MACHINE LEARNING SYSTEM AND MACHINE LEARNING METHOD INVOLVING DATA AUGMENTATION, AND STORAGE MEDIUM**

MASCHINENLERNVERFAHREN UND MASCHINENLERNSYSTEM MIT DATENERWEITERUNG, UND SPEICHERMEDIUM

PROCÉDÉ D'APPRENTISSAGE AUTOMATIQUE ET SYSTÈME D'APPRENTISSAGE AUTOMATIQUE IMPLIQUANT UNE AUGMENTATION DE DONNÉES, ET SUPPORT DE STOCKAGE

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR

(30) Priority: **05.06.2020 US 202063034993 P**

(43) Date of publication of application:
08.12.2021 Bulletin 2021/49

(73) Proprietor: **HTC Corporation**
Taoyuan City 330 (TW)

(72) Inventors:
- **CHEN, Chih-Yang**
  330 Taoyuan City (TW)
- **CHANG, Che-Han**
  330 Taoyuan City (TW)
- **CHANG, Edward**
  330 Taoyuan City (TW)

(74) Representative: **Winter, Brandl - Partnerschaft mbB**
Alois-Steinecker-Straße 22
85354 Freising (DE)

(56) References cited:
- **FELIPE PETROSKI SUCH ET AL: "Generative Teaching Networks: Accelerating Neural Architecture Search by Learning to Generate Synthetic Training Data", ARXIV.ORG, CORNELL UNIVERSITY LIBRARY, 201 OLIN LIBRARY CORNELL UNIVERSITY ITHACA, NY 14853, 17 December 2019 (2019-12-17), XP081560001**
- **HA DAVID ET AL: "Hypernetworks", 1 December 2016 (2016-12-01), XP055773067, Retrieved from the Internet <URL:https://arxiv.org/pdf/1609.09106.pdf> [retrieved on 20210208]**
- **MATTHEW MACKAY ET AL: "Self-Tuning Networks: Bilevel Optimization of Hyperparameters using Structured Best-Response Functions", ARXIV.ORG, CORNELL UNIVERSITY LIBRARY, 201 OLIN LIBRARY CORNELL UNIVERSITY ITHACA, NY 14853, 7 March 2019 (2019-03-07), XP081130759**

EP 3 920 102 B1

Description

BACKGROUND

Field of Invention

[0001] The disclosure relates to a machine learning system, a machine learning method and a storage medium. More particularly, the disclosure relates to a machine learning system and a machine learning method involving data augmentation function.

Description of Related Art

[0002] Technologies such as machine learning and neural networks are widely used in a technical field of computer vision. One of the important applications of computer vision is to detect or identify objects (such as human faces, vehicle license plates, etc.) contained in pictures or images. The object detection can be realized through feature extraction and feature classification.

[0003] In order to correctly detect objects in pictures or images and improve the accuracy of detection, it requires a large amount of training data (such as input images and corresponding classification labels attached to the input images for training), so that the neural network for classification is able to learn a correlation between the input image and the correct classification label from the training data. In practice, it is quite difficult to obtain a sufficient amount of training data to meet the accuracy requirements. Lack of sufficient training data samples becomes a common problem among various object detection applications.

[0004] In this respect, Felipe Petroski Such et. al.; "Generative Teachings Networks: Accelerating Neural Architecture Search by Learning to Generate Synthetic Training Date", Arxiv.org, Cornell University Library, 201 Online Library Cornell University Ithaca, NY 14853, 17. December 2019, discloses a method for producing synthetic training data and/or training environments for a learner, e.g. a freshly initialized neural network, by using deep neural networks, so-called Generative Teachings Networks (GTNs). Furthermore, David Ha et. al.; "HyperNetworks", http://arxiv.org/pdf/1609.09106.pdf; 1. December 2016, discloses Hypernetworks which are used to generate weights for another network. Matthew MacKay et. al., "Self-Tuning Networks: Bilevel Optimization of Hyperparameters using structured best-response Functions"; Arxiv.org, Cornell University Library, 201 Online Library Cornell University Ithaca, NY 14853, 7. March 2019, discloses a method for hyperparameter optimization.

[0005] Starting from Felipe Petroski Such et.al., the object of the present invention is to improve the accuracy of a classification model and to reduce an over-fitting issue such that the overall model performance is improved. This object is solved by a machine learning system according to claim 1, a machine learning method according to claim 7 and a non-transitory computer-readable storage medium according to claim 15. Preferred embodiments are subject matter of the dependent claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] The disclosure can be more fully understood by reading the following detailed description of the embodiment, with reference made to the accompanying drawings as follows:

FIG. 1 is a schematic diagram illustrating a machine learning system according to an embodiment of the disclosure.

FIG. 2 is a schematic diagram illustrating a machine learning method according to an embodiment of the disclosure.

FIG. 3 is a flowchart illustrating further steps within one step shown in FIG. 2 in some embodiments.

FIG. 4 is a schematic diagram illustrating steps performed by components of the processing unit in some embodiments.

FIG. 5A is a schematic diagram illustrating a conversion from a hyperparameter into the first classification model parameter by the hypernetwork based on the hypernetwork parameter according to some embodiments of the disclosure.

FIG. 5B is a schematic diagram illustrating the hypernetwork parameter updated according to the first loss according to some embodiments of the disclosure.

FIG. 6 is a schematic diagram illustrating an internal structure of the four exploration classification models formed from the classification model based on four exploration classification model parameters according to some embodiments of the disclosure.

FIG. 7 is a flowchart illustrating detailed steps within one step shown in FIG. 2 in some embodiments.

FIG. 8 is a schematic diagram illustrating steps performed by components of the processing unit in some embodiments.

FIG. 9A is a schematic diagram illustrating a conversion from the hyperparameter into the second classification model parameter in some embodiments of the disclosure.

FIG. 9B is a schematic diagram illustrating the updating of the hyperparameter according to the sec-

ond loss in some embodiments of the disclosure.

DETAILED DESCRIPTION

[0007] Reference is made to FIG. 1, which is a schematic diagram illustrating a machine learning system 100 according to an embodiment of the disclosure. The machine learning system 100 includes a memory unit 120 and a processing unit 140. The processing unit 140 is coupled with the memory unit 120.

[0008] In some embodiments, the machine learning system 100 can be established by a computer, a server or a processing center. In some embodiments, the processing unit 140 can be realized by a processor, a central processing unit or a computing unit. In some embodiments, the memory unit 120 can be realized by a memory, a flash memory, a read-only memory (ROM), a hard disk or any equivalent storage component.

[0009] In some embodiments, the machine learning system 100 is not limited to include the memory unit 120 and the processing unit 140. The machine learning system 100 may further include other components required to operating the machine learning system 100 in various applications. For example, the machine learning system 100 may further include an output interface (e.g., a display panel for displaying information), an input interface (e.g., a touch panel, a keyboard, a microphone, a scanner or a flash memory reader) and a communication circuit (e.g., a WiFi communication module, a Bluetooth communication module, a wireless telecommunication module, etc.).

[0010] As shown in FIG. 1, initial values of at least two parameters, which include a hyperparameter HP and a hypernetwork parameter HNP, are stored in the memory unit 120. The machine learning system 100 decides how to perform data augmentation and label classification based on these two parameters (i.e., the hyperparameter HP and the hypernetwork parameter HNP), and further details will be discussed in following paragraphs. Data augmentation is a technology to increase the amount of training data. While training a machine learning model (or a machine learning model), it usually requires a lot of training data. By applying data augmentation to original training data, the original training data can be expanded to a larger amount of augmented training data, so as to avoid an over-fitting issue while training the deep learning model (or the machine learning model).

[0011] As shown in FIG. 1, the processing unit 140 is coupled with the memory unit 120. The processing unit 140 is configured to run a data augmentation model 142, a hypernetwork 144 and a classification model 146 based on corresponding software/firmware instruction programs.

[0012] The data augmentation model 142 is configured to perform data augmentation on an inputted training sample to generate multiple augmented training samples. For example, when the inputted training sample includes one original image (e.g., a photo with a car running on a roadway in a daytime) and a training label corresponding to the original image (e.g., car, road or traffic light). The data augmentation model 142 is configured to perform a combination of one or more processes among horizontally flip, vertically flip, rotate, vertically shift, horizontally shift, zoom-in, zoom-out and brightness adjustment on the original image.

[0013] The data augmentation model 142 processes the original image with different settings (e.g., applying different rotation angles or different zoom-in/zoom-out ratios) based on values of the hyperparameter HP to generate multiple data augmentation images of the multiple augmented training samples. Even though these data augmentation images are generated according to the original image, the pixel values in the data augmentation images are changed because of image processing. To the classification model 146, these data augmentation images are equivalent to different training samples, so as to extend the amount of the training samples and solve the insufficiency of the training samples.

[0014] The classification model 146 can classify the input data (such as the aforementioned data augmentation images), for example, detecting that the input image contains vehicles, faces, license plates, text, totems, or other image-feature objects. The classification model 146 is configured to generate a corresponding label according to a classification result. It should be noted that the classification model 146 will refer to a classification model parameter while performing classification operations.

[0015] The hypernetwork 144 is configured to convert the hyperparameter HP into the classification model parameter used by the classification model 146. The hypernetwork 144 determines how to convert the hyperparameter HP into the classification model parameters according to the hypernetwork parameter HNP.

[0016] In other words, the hyperparameter HP determines how the data augmentation model 142 performs data augmentation, and also the hyperparameter HP is transformed by the hypernetwork 144 (into the classification model parameter) to determine how the classification model 146 performs classification operations.

[0017] Reference is further made to FIG. 2, which is a schematic diagram illustrating a machine learning method 200 according to an embodiment of the disclosure. The machine learning system 100 shown in FIG. 1 can be utilized to perform the machine learning method 200 shown in FIG. 2.

[0018] As shown in FIG. 2, firstly in step S210, the initial values of the hyperparameter HP and the hypernetwork parameter HNP are obtained. In some embodiments, the initial values of the hyperparameter HP and the hypernetwork parameter HNP can be obtained according to average values from historical training practices, manual-setting default values, or random values.

[0019] In step S220, the first classification model parameter is generated according to the hyperparameter and the hypernetwork parameter, and the hypernetwork

parameters are updated based on a classification result about a training sample based on the first classification model parameter. The hypernetwork 144 (based on the hypernetwork parameter HNP) converts the hyperparameter HP into the first classification model parameter, and the hypernetwork parameter HNP is updated according to the classification result relative to the training sample based on the first classification model parameter. Further details about step S220 will be further described in following paragraphs with some examples.

[0020] In step S230, the second classification model parameters are generated according to the hyperparameter and the updated hypernetwork parameter, and the hyperparameters are updated according to another classification result about a verification sample based on the second classification model parameter. The hypernetwork 144 (based on the updated hypernetwork parameter HNP) converts the hyperparameter HP into the second classification model parameter, and the hyperparameter HP is updated according to the another classification result about the verification sample based on the second classification model parameter. Further details about step S230 will be further described in following paragraphs with some examples.

[0021] In other words, in step S220, the hypernetwork parameter HNP is updated first. Then, in step S230, the hyperparameter HP is updated based on the new hypernetwork parameter HNP.

[0022] In step S240, it is to determine whether a convergence condition is fulfilled. If the convergence condition has not been fulfilled, it returns to step S220 again, and continues to repeat steps S220 and S230 for updating the hypernetwork parameter HNP and the hyperparameter HP. Before the convergence condition is fulfilled, steps S220 and S230 are performed repeatedly for gradually updating the hypernetwork parameter HNP and the hyperparameter HP in an iterative manner.

[0023] If the convergence condition has been fulfilled (for example, an accuracy of the classification result given by the classification model 146 exceeds a threshold, a number of training rounds reaches a predetermined number of rounds, am amount of training samples reaches a predetermined amount of samples, or a time length of training duration reaches the predetermined time length, etc.), it means that the machine learning system 100 has completed the training, and the classification model 146 after training can be used to execute subsequent applications. For example, the classification model 146 after the training can be used for object recognition, face recognition, audio recognition, or motion detection within input pictures, images or streaming data.

[0024] Reference is further made to FIG. 3 and FIG. 4. FIG. 3 is a flowchart illustrating further steps S221 to S225 within step S220 in some embodiments. FIG. 4 is a schematic diagram illustrating steps S221 to S225 performed by components of the processing unit 140 in some embodiments.

[0025] As shown in FIG. 4, it is assumed that in an initial state, the initial value of the hyperparameter is the hyperparameter HP1, and the initial value of the hypernetwork parameter is the hypernetwork parameter HNP1.

[0026] As shown in FIG. 3 and FIG. 4, in step S221, the data augmentation model 142 performs data augmentation on the training sample TD based on the hyperparameter HP1 to generate an augmented training sample ETD. In step S222, the hypernetwork 144 converts the hyperparameter HP1 into the first classification model parameter MP1 based on the hypernetwork parameter HNP1.

[0027] Reference is further made to FIG. 5A, which is a schematic diagram illustrating a conversion from the hyperparameter HP1 into the first classification model parameter MP1 by the hypernetwork 144 based on the hypernetwork parameter HNP1 in step S222. As shown in FIG. 5A, step S222 is executed to map a data point (i.e., hyperparameter HP1) in data augmentation space SP1 to a data point (i.e., the first classification model parameter MP1) in classification parameter space SP2.

[0028] In FIG. 5A, the data augmentation space SP1, for demonstration, is a plane coordinate system with two axes. For example, one axis can represent a rotation angle during data augmentation, and the other axis can represent a ratio of size scaling during data augmentation. In this case, the data points located at different positions of the data augmentation space SP1 correspond to different settings of data augmentation. The classification parameter space SP2, for demonstration, is a three-dimensional coordinate system with three axes, and the three axes can respectively represent three weight values of convolutional layers (in the classification model). In step S222, the hypernetwork parameter HNP1 is used to determine how the hypernetwork 144 maps the hyperparameter HP1 within the data augmentation space SP1 onto the first classification model parameter MP1 within the classification parameter space SP2. If the hypernetwork parameter HNP1 changes, the hypernetwork 144 will map the hyperparameter HP1 onto another position within the classification parameter space SP2.

[0029] It is added that, for brevity of description, the data augmentation space SP1 and the classification parameter space SP2 in FIG. 5A are illustrated with two axes and three axes respectively for demonstration. The disclosure is not limited thereto. In practical applications, the data augmentation space SP1 and the classification parameter space SP2 may have different dimension configurations. In some embodiments, the classification parameter space SP2 is a high-dimensional space with more axes.

[0030] As shown in FIG. 3 and FIG. 4, in step S223, the classification model 146 classifies the augmented training sample ETD based on the first classification model parameter MP1 to generate a first predicted label LPD1 corresponding to the augmented training sample ETD.

[0031] In step S224, the processing unit 140 executes a comparison algorithm for comparing the first prediction

label LPD1 with a training label LTD of the training sample TD to generate a first loss L1. In some embodiments, the processing unit 140 performs a cross-entropy calculation on the first predicted label LPD1 and the training label LTD to obtain the first loss L1.

[0032] A value of the first loss L1 represents whether the classification result performed by the classification model 146 is accurate. If the first prediction label LPD1 generated by the classification model 146 is the same (or similar) to the training label LTD of the training sample TD, the value of the first loss of L1 will be small, and it means that the first classification model parameter MP1 currently adopted by the classification model 146 is more accurate. If the first prediction label LPD1 generated by the classification model 146 is different from the training label LTD of the training sample TD, the value of the first loss L1 will be larger, and it means that the first classification model parameter MP1 currently adopted by the classification model 146 is relatively inaccurate.

[0033] In step S225, the hypernetwork parameter HNP2 is updated according to the first loss L1. Reference is further made to FIG. 5B, which is a schematic diagram illustrating the hypernetwork parameter HNP2 updated according to the first loss L1 in step S225 according to some embodiments of the disclosure. As shown in Fig. 5B, after obtaining the first loss L1 corresponding to the first classification model parameter MP1 currently adopted by the classification model 146, the first loss L1 is backward propagated to the classification model 146, so as to obtain an improved classification model parameter MP1m which can reduce (or minimize) the first loss L1. Then, the improved classification model parameter MP1m is backward propagated to the hypernetwork 144, and an updated hypernetwork parameter HNP2 is obtained according to the backpropagation based on the improved classification model parameter MP1m. In some embodiments, stochastic gradient descent (SGD) algorithm can be used to find the improved classification model parameter MP1m to reduce (or minimize) the first loss L1.

[0034] As shown in FIG. 4 and FIG. 5B, under the condition that the hyperparameter HP1 remains the same, the hypernetwork 144 (based on the updated hypernetwork parameter HNP2) will map the hyperparameter HP1 onto the improved classification model parameter MP1m.

[0035] In some embodiments, as shown in FIG. 5A, a plurality of exploration values are introduced in step S222, and these exploration values are used to form a plurality of exploration hyperparameters around the hyperparameter HP1, and each of the exploration values include slight difference in an axis (for example, the rotation angle increases/decreases by 0.5 degrees, the shifting distance increases/decreases by 1%, etc.). As shown in FIG. 5A, there are four exploration hyperparameters HPe1~HPe4 located around the hyperparameter HP1. In addition to mapping the hyperparameter HP1 onto the first classification model parameter MP1 in the classification parameter space SP2, the hypernetwork 144 (based on the hypernetwork parameter HNP1) will map the exploration hyperparameters HPe1~HPe4 formed by these exploration values onto four exploration classification model parameters MPe1~MPe4 in the classification parameter space SP2. In FIG. 5A, the exploration classification model parameters MPe1~MPe4 are also adjacent to the original first classification model parameter MP1. In some embodiments, the first classification model parameter MP1 can also be regarded as one of the exploration classification model parameters.

[0036] In other words, when four exploration hyperparameters are added, the four exploration hyperparameters HPe1~HPe4 will be mapped to the other four exploration classification model parameters MPe1~MPe4. The amount of aforementioned exploration hyperparameters (i.e., four exploration hyperparameters) is given for demonstration, and the amount of exploration hyperparameters is not limited to four in practical applications.

[0037] In some embodiments, four exploration classification models will be generated according to the four exploration classification model parameters MPe1~MPe4, and the four exploration classification models will classify the training sample TD respectively and produce four outcomes of the first prediction labels LPD1. In step S224, the four outcomes of the first prediction labels LPD1 are compared with the training label LTD respectively, and correspondingly it will obtain four outcomes of the first losses L1 corresponding to the four exploration classification models respectively. In some embodiments, the four outcomes of the first prediction labels LPD1 and the training label LTD are compared by cross-entropy calculation respectively for obtaining the first losses L1.

[0038] In this embodiment, in step S225, the four exploration classification models and the four outcomes of the first losses L1 can be all taken in consideration while updating the hypernetwork parameter HNP1 into the hypernetwork parameter HNP2.

[0039] Reference is further made to FIG. 6, which is a schematic diagram illustrating an internal structure of the four exploration classification models 146e1~146e4 formed from the classification model 146 based on four exploration classification model parameters MPe1~MPe4 according to some embodiments of the disclosure. As shown in FIG. 6, each of the exploration classification models 146e1~146e4 includes n neural network structure layers SL1, SL2, SL3, SL4, SL5... SLn. In some embodiments, each of the neural network structure layers SL1, SL2, SL3, SL4, SL5...SLn can be a convolution layer, a pooling layer, a linear rectification layer, a fully connected layer or other type of neural network structure layer.

[0040] In some embodiments, n is a positive integer. In general, the total number of layers in the classification model can be determined according to application requirements (e.g., classification accuracy requirement, complexity of classification target, and diversity of input images). In some cases, a common range of n can be

ranged between 16 and 128, and the disclosure is not limited to a specific number of layers.

[0041] For example, the neural network structure layers SL1 and SL2 can be convolutional layers; the neural network structure layer SL3 can be a pooling layer; the neural network structure layers SL4 and SL5 can be convolutional layers; the neural network structure layer SL6 can be a pooling layer, the neural network structure layer SL7 can be a convolutional layer; the neural network structure layer SL8 can be a linear rectification layer; and the neural network structure layer SLn can be a fully connected layer, and the disclosure is not limited thereto.

[0042] As shown in FIG. 6, the neural network structure layers SL1 ~SLn are divided into a first structure layer portion P1 and a second structure layer portion P2 after the first structure layer portion P1. In the embodiment shown in FIG. 6, the first structure layer portion P1 includes the neural network structure layers SL1~SL3, and the second structure layer portion P2 includes the neural network structure layers SL4~SLn.

[0043] Each one of the exploration classification model parameters MPe1~MPe4 for forming the exploration classification models 146e1~146e4 includes a first weight parameter content (configured to determine the operation of the first structural layer portion P1) and a second weight parameter content (configured to determine the operation of the second structure layer portion P2). In some embodiments, the second structure layer portions P2 (i.e., the neural network structure layers SL4-SLn) of the four exploration classification models 146e1~146e4 share the same second weight parameter content, and the neural network structure layers SL4~SLn among the four exploration classification models 146e1~146e4 are operating with the same logic.

[0044] In other words, the neural network structure layer SL4 of the exploration classification model 146e1 and the neural network structure layer SL4 of the exploration classification model 146e2 use the same weight parameters and are operating with the same logic. Similarly, the neural network structure layer SL5 of the exploration classification model 146e1 and the neural network structure layer SL5 of the exploration classification model 146e2 use the same weight parameters and are operating with the same logic, and so on.

[0045] On the other hand, each one of the first structure layer portions P1 (i.e., the neural network structure layers SL1~SL3) of the four exploration classification models 146e1~146e4 has the first weight parameter content independent from others. The logic of the neural network structure layer SL1~SL3 in one exploration classification model is different from the logic of the neural network structure layer SL1~SL3 in another exploration classification model.

[0046] The distribution of the first structure layer portion P1 and the second structure layer portion P2 shown in FIG. 6 is for demonstration, and the disclosure document is not limited thereto.

[0047] In an embodiment, the first structure layer portion P1 in each of the exploration classification models 146e1~146e4 at least includes a first convolutional layer. For example, the first structure layer part P1 includes the neural network structure layer SL1 (i.e., the first convolutional layer), the first convolutional layers of the exploration classification models 146e1~ 146e4 have different weight parameters from each other. In this embodiment, the rest of the neural network structure layers SL2~SLn all belong to the second structure layer portion P2 (not shown in the figure), and the second structure layer part P2 includes a second convolutional layer and a fully connected layer. The second convolutional layer and the fully connected layer of the exploration classification models 146e1~146e4 have the same weight parameters across the classification models 146e1~146e4. In this embodiment, since most of the neural network structure layers SL2~SLn share the same weight parameters, only fewer neural network structure layer (e.g., the neural network structure layer SL1) uses independent weight parameters, the neural network structure is relatively simple while training, able to achieve a faster training speed, requires less computing resources, and also able to maintain accuracy according to experiment outcomes.

[0048] Reference is further made to FIG. 7 and FIG. 8. FIG. 7 is a flowchart illustrating detailed steps S231 to S234 within step S230 in some embodiments. FIG. 8 is a schematic diagram illustrating steps S231 to S234 performed by components of the processing unit 140 in some embodiments.

[0049] After step S220 shown in FIG. 3 and FIG. 4, when the method enters step S230, as shown in FIG. 8, the current value of the hyperparameter is still the hyperparameter HP1, and the current value of the hypernetwork parameter has been updated to the hypernetwork parameter HNP2.

[0050] As shown in FIG. 7 and FIG. 8, in step S231, the hypernetwork 144 (based on the updated hypernetwork parameter HNP2) converts the hyperparameter HP1 into the second classification model parameter MP2. The classification model parameter MP2 is equal to the improved classification model parameter MP1m obtained in the aforesaid embodiments in FIG. 5B through back-propagation. Reference is further made to FIG. 9A, which is a schematic diagram illustrating a conversion from the hyperparameter HP1 into the second classification model parameter MP2 in step S231 in some embodiments of the disclosure. As shown in FIG. 9A, step S231 is configured to map a data point (i.e., the hyperparameter HP1) in the data augmentation space SP1 onto a data point (i.e., the second classification model parameter MP2) in the classification parameter space SP2.

[0051] In step S231, the hypernetwork parameter HNP2 is used to determine how the hypernetwork 144 maps the hyperparameter HP1 in the data augmentation space SP1 onto the second classification model parameter MP2 in the classification parameter space SP2.

[0052] Comparing FIG. 9A with FIG. 5A, since the hypernetwork parameter HNP2 is already different from the

hypernetwork parameter HNP1 in the previous embodiment (as shown in FIG. 5A), the same hyperparameter HP1 will be mapped by the hypernetwork 144 onto a new position (i.e., the second classification model parameter MP2) in the classification parameter space SP2.

[0053] As shown in FIG. 7 and FIG. 8, in step S232, the classification model 146 classifies a verification sample VD based on the second classification model parameter MP2 to generate a second prediction label LPD2 corresponding to the verification sample VD.

[0054] In step S233, the processing unit 140 executes a comparison algorithm to compare the second predicted label LPD2 with the verification label LVD of the verification sample VD for generating a second loss L2. In some embodiments, the processing unit 140 performs cross-entropy calculation between the second predicted label LPD2 and the verification label LVD to obtain the second loss L2.

[0055] A value of the second loss L2 represents whether the classification result performed by the classification model 146 is accurate. If the second prediction label LPD2 generated by the classification model 146 is the same (or similar) to the verification label LVD of the verification sample VD, the value of the second loss will be small, and it means that the second classification model parameter MP2 adopted by the current classification model 146 is more accurate. If the second prediction label LPD2 generated by the classification model 146 is different from the verification label LVD of the verification sample VD, the value of the second loss L2 will be larger, and it means that the second classification model parameter MP2 adopted by the current classification model 146 is relatively inaccurate.

[0056] In step S234, the hyperparameter HP1 is updated into the hyperparameter HP2 according to the second loss L2. Reference is further made to FIG. 9B, which is a schematic diagram illustrating the updating of the hyperparameter HP2 according to the second loss L2 in step S234 in some embodiments of the disclosure. As shown in Fig. 9B, after obtaining the second loss L2 corresponding to the second classification model parameter MP2 currently adopted by the classification model 146, the second loss L2 is backward propagated to the classification model 146, so as to obtain an improved classification model parameter MP2m which can reduce (or minimize) the second loss L2. Then, the improved classification model parameter MP2m is backward propagated to the hypernetwork 144, and an updated hyperparameter HP2 is obtained according to the backpropagation based on the improved classification model parameter MP2m. In some embodiments, stochastic gradient descent (SGD) algorithm can be used to find the improved classification model parameter MP2m to reduce (or minimize) the second loss L2.

[0057] As shown in FIG. 8 and FIG. 9B, if the hypernetwork parameter HNP2 used by the hypernetwork 144 remains unchanged, the hypernetwork 144 (based on the hypernetwork parameter HNP2) will map the updated hyperparameter HP2 onto the improved classification model parameter MP2m.

[0058] Based on aforesaid embodiments, in step S220, firstly, the hypernetwork parameter HNP1 is updated to the hypernetwork parameter HNP2. In step S230, the hyperparameter HP1 is updated to the hyperparameter HP2 based on the hypernetwork parameter HNP2. When step S230 is completed, if the convergence condition is not fulfilled yet, the method returns to step S220 based on the hyperparameter HP2, and perform steps S220 and S230 again with the hyperparameter HP2 and the hypernetwork parameter HNP2 as input conditions. In this case, the hyperparameters and hyperparameters can be updated again, and so on. The hypernetwork parameters and hyperparameters can be updated iteratively until the convergence conditions are fulfilled.

[0059] As shown in FIG. 1, during the training process of the machine learning system 100, the hyperparameter HP is configured to control the data augmentation operation of the data augmentation model 142, and the hyperparameter HP (through transformation by the hypernetwork 144) is also configured to control the classification operation of the classification model 146. In addition, different exploration classification models in the disclosure can share weights. By sharing weights, it can save storage and computing resources and accelerate the training speed. In addition, the machine learning system 100 of the disclosure may utilize the data augmentation model to increase the equivalent number of training samples TD, such that the training process of the classification model 146 does not require a large number of training samples TD, and the classification model 146 can still maintain high accuracy.

[0060] In the field of computer vision, the accuracy of deep learning mainly relies on a large amount of labeled training data. As the quality, quantity, and variety of training data increase, the performance of the classification model usually improves correspondingly. However, it is difficult to collect high-quality data to train the classification model. Therefore, it is hard to improve the performance of the classification model. One of the ways to solve this problem is to allow experts to manually design parameters for data augmentation, such as rotation angle, flip method, or brightness adjustment ratio. The data augmentation with manually designed parameters has been commonly used to train the high-performance classification model for computer vision. If machine learning can be used in automatically finding the parameters for data augmentation, it will be more efficient and more accurate. In aforesaid embodiments of the disclosure, it proposes a hypernetwork-based data augmentation (HBA), which generates multiple continuous exploration models using the hypernetwork, and uses the gradient descent method to automatically adjust the hyperparameters for data augmentation. Some embodiments of the disclosure adopt a weight sharing strategy to improve the speed and accuracy of calculation, and it can save time and resources

for manually adjusting the parameters for data augmentation. In addition, whether the original training samples are sufficient or not, the data augmentation can effectively improve the accuracy of the classification model and reduce the over-fitting issue. Therefore, automatic adjustment of parameters for data augmentation can improve the overall model performance.

[0061] For practical applications, the machine learning method and the machine learning system in the disclosure can be utilized in various fields such as machine vision, image classification, or data classification. For example, this machine learning method can be used in classifying medical images. The machine learning method can be used to classify X-ray images in normal conditions, with pneumonia, with bronchitis, or with heart disease. The machine learning method can also be used to classify ultrasound images with normal fetuses or abnormal fetal positions. On the other hand, this machine learning method can also be used to classify images collected in automatic driving, such as distinguishing normal roads, roads with obstacles, and road conditions images of other vehicles. The machine learning method can be utilized in other similar fields. For example, the machine learning methods and machine learning systems in the disclosure can also be used in music spectrum recognition, spectral recognition, big data analysis, data feature recognition and other related machine learning fields.

[0062] Another embodiment in the disclosure is a non-transitory computer-readable medium containing at least one instruction program, which is executed by a processor (for example, the processing unit 140 in FIG. 1) to perform the machine learning method 200 in the embodiments shown in FIG. 2, FIG. 3, and FIG. 7.

Claims

1. A machine learning system (100), comprising:

a memory unit (120), configured for storing initial values of a hyperparameter (HP) and a hypernetwork parameter (HNP);
a processing unit (140), coupled with the memory unit, wherein the processing unit is configured to run a hypernetwork (144), a classification model (146), which is configured to classify an image by detecting that the image contains one or more objects and to generate a corresponding label according to a classification result, and a data augmentation model (142), the processing unit is configured to execute operations of:

(a) (S220) generating a first classification model parameter (MP1) by the hypernetwork according to the hyperparameter (HP1) and the hypernetwork parameter (HNP1), generating a classification result by the classification model based on the first classification model parameter relative to a training sample (TD), and updating the hypernetwork parameter (HNP1, HNP2) according to the classification result, wherein the operation (a) executed by the processing unit comprises:

(a1) (S221) performing data augmentation, by the data augmentation model based on the hyperparameter, on the training sample for generating an augmented training sample (ETD), wherein the data augmentation model is utilized to perform a combination of one or more image processes on the training sample based on values of the hyperparameter to generate the augmented training sample;
(a2) (S222) converting the hyperparameter, by the hypernetwork based on the hypernetwork parameter, into the first classification model parameter, wherein the hypernetwork is configured to map a data point of the hyperparameter in a data augmentation space (SP1) to a data point of the first classification model parameter in a classification parameter space (SP2) based on the hypernetwork parameter, wherein the data augmentation space (SP1) is defined by axes representing different settings during data augmentation, wherein the classification parameter space (SP2) is defined by axes representing weight values of convolutional layers in the classification model;
(a3) (S223) performing classification, by the classification model based on the first classification model parameter, on the augmented training sample for generating a first prediction label (LPD1) corresponding to the augmented training sample, wherein the first classification model parameter comprises first weight values of convolutional layers in the classification model; and
(a4) (S224, S225) updating the hypernetwork parameter according to a first loss (L1) generated by comparing the first prediction label with a training label (LVD) of the training sample;

(b) (S230) generating a second classification model parameter (MP2) by the hypernetwork according to the hyperparameter (HP1) and the updated hypernetwork parameter (HNP2), generating another classi-

fication result by the classification model based on the second classification model parameter relative to a verification sample (VD), and updating the hyperparameter (HP1, HP2) according to the another classification result, wherein the second classification model parameter comprises second weight values of the convolutional layers in the classification model, and at least a part of the second weight values are different from the first weight values; and
(c) (S240) repeating the operations (a) and (b) for updating the hypernetwork parameter and the hyperparameter.

2. The machine learning system of claim 1, wherein the operation (a2) executed by the processing unit comprises:

converting the hyperparameter, by the hypernetwork based on the hypernetwork parameter and a plurality of exploration values, into a plurality of exploration classification model parameters (MPe1, MPe2, MPe3, MPe4);
wherein the operation (a3) executed by the processing unit comprises:

forming a plurality of exploring classification models (146e1, 146e2, 146e3, 146e4) by the classification model based on the exploration classification model parameters respectively, and performing classification on the augmented training sample by the exploring classification models respectively for generating a plurality of first prediction labels (LPD1) corresponding to the augmented training sample; and
wherein the operation (a4) executed by the processing unit comprises:

calculating a plurality of first losses (L1) by comparing the first prediction labels with the training label of the training sample; and
updating the hypernetwork parameter (HNP1, HNP2) according to the exploring classification models and the first losses corresponding to the exploring classification models.

3. The machine learning system of claim 2, wherein each of the exploring classification models (146e1, 146e2, 146e3, 146e4) comprises a plurality of neural network structural layers (SL1~SLn), the neural network structural layers are divided into a first structural layer portion (P1) and a second structural layer portion (P2) after the first structural layer portion, each of the exploration classification model parameters for forming the exploring classification models comprises a first weight parameter content and a second weight parameter content, the first weight parameter content is configured to determine operations of the first structural layer portion, and the second weight parameter content is configured to determine operations of the second structural layer portion.

4. The machine learning system of claim 3, wherein the second weight parameter contents applied to the second structural layer portions of the exploring classification models are the same, the second structural layer portions of the exploring classification models are operating with the same logic.

5. The machine learning system of claim 3, wherein the first structural layer portion in each of the exploring classification models comprises at least one first convolutional layer (SL1), the first convolutional layers among the exploring classification models have different weight parameters from each other, the second structural layer portion in each of the exploring classification models comprises at least one second convolutional layer (SL4) and at least one fully connection layer (SLn), the second convolutional layers and the fully connection layers among the exploring classification models have same weight parameters across the exploring classification models.

6. The machine learning system of claim 1, wherein the operation (b) (S230) executed by the processing unit comprises:

(b1) (S231) converting the hyperparameter (HP1), by the hypernetwork based on the updated hypernetwork parameter (HNP2), into the second classification model parameter (MP2);
(b2) (S232) performing classification, by the classification model based on the second classification model parameter, on the verification sample for generating a second prediction label (LPD2) corresponding to the verification sample; and
(b3) (S233, S234) updating the hyperparameter (HP1, HP2) according to a second loss (L2) generated by comparing the second prediction label with a verification label (LVD) of the verification sample.

7. A machine learning method (200), wherein the machine learning method is performed by a machine learning system (100) comprising a processing unit (140), the machine learning method comprising:

(a) (S210) obtaining initial values of a hyperparameter (HP) and a hypernetwork parameter (HNP);
(b) (S220) generating a first classification model

parameter (MP1) according to the hyperparameter (HP1) and the hypernetwork parameter (HNP1), generating a classification result by a classification model, which is configured to classify an image by detecting that the image contains one or more objects and to generate a corresponding label according to a classification result, based on the first classification model parameter relative to a training sample (TD), and updating the hypernetwork parameter (HNP1, HNP2) according to a classification result based on the first classification model parameter relative to a training sample by the processing unit (140), wherein the step (b) comprises:

(b1) (S221) performing data augmentation, by a data augmentation model based on the hyperparameter, on the training sample for generating an augmented training sample (ETD), wherein the data augmentation model is utilized to perform a combination of one or more image processes on the training sample based on values of the hyperparameter to generate the augmented training sample;
(b2) (S222) converting the hyperparameter, by a hypernetwork based on the hypernetwork parameter, into the first classification model parameter, wherein the hypernetwork is configured to map a data point of the hyperparameter in a data augmentation space (SP1) to a data point of the first classification model parameter in a classification parameter space (SP2) based on the hypernetwork parameter, wherein the data augmentation space (SP1) is defined by axes representing different settings during data augmentation, wherein the classification parameter space (SP2) is defined by axes representing weight values of convolutional layers in the classification model;
(b3) (S223) performing classification, by the classification model based on the first classification model parameter, on the augmented training sample for generating a first prediction label (LPD1) corresponding to the augmented training sample, wherein the first classification model parameter comprises first weight values of convolutional layers in the classification model; and
(b4) (S224, S225) updating the hypernetwork parameter according to a first loss (L1) generated by comparing the first prediction label with a training label (LVD) of the training sample;

(c) (S230) generating a second classification model parameter (MP2) according to the hyperparameter (HP1) and the updated hypernetwork parameter (HNP2), generating another classification result by the classification model based on the second classification model parameter relative to a verification sample (VD), and updating the hyperparameter (HP1, HP2) according to the another classification result, wherein the second classification model parameter comprises second weight values of the convolutional layers in the classification model, and at least a part of the second weight values are different from the first weight values; and
(d) (S240) repeating the steps (b) and (c) for updating the hypernetwork parameter and the hyperparameter.

8. The machine learning method of claim 7, wherein the step (b2) comprises:

converting the hyperparameter, by the hypernetwork based on the hypernetwork parameter and a plurality of exploration values, into a plurality of exploration classification model parameters;
wherein the step (b3) comprises:

forming a plurality of exploring classification models (146e1, 146e2, 146e3, 146e4) by the classification model based on the exploration classification model parameters respectively, and performing classification on the augmented training sample by the exploring classification models respectively for generating a plurality of first prediction labels (LPD1) corresponding to the augmented training sample; and
wherein the step (b4) comprises:

calculating a plurality of first losses (L1) by comparing the first prediction labels with the training label of the training sample; and
updating the hypernetwork parameter (HNP1, HNP2) according to the exploring classification models and the first losses corresponding to the exploring classification models.

9. The machine learning method of claim 8, wherein in the step (b4):
calculating the first losses by a cross-entropy calculation between the first prediction labels of the exploring classification models and the training label respectively.

10. The machine learning method of claim 8, wherein each of the exploring classification models (146e1, 146e2, 146e3, 146e4) comprises a plurality of neural

network structural layers (SL1~SLn), the neural network structural layers are divided into a first structural layer portion (P1) and a second structural layer portion (P2) after the first structural layer portion, each of the exploration classification model parameters for forming the exploring classification models comprises a first weight parameter content and a second weight parameter content, the first weight parameter content is configured to determine operations of the first structural layer portion, and the second weight parameter content is configured to determine operations of the second structural layer portion.

11. The machine learning method of claim 10, wherein the second weight parameter contents applied to the second structural layer portions of the exploring classification models are the same, the second structural layer portions of the exploring classification models are operating with the same logic.

12. The machine learning method of claim 10, wherein the first structural layer portion in each of the exploring classification models comprises at least one first convolutional layer (SL1), the first convolutional layers among the exploring classification models have different weight parameters from each other, the second structural layer portion in each of the exploring classification models comprises at least one second convolutional layer (SL4) and at least one fully connection layer (SLn), the second convolutional layers and the fully connection layers among the exploring classification models have same weight parameters across the exploring classification models.

13. The machine learning method of claim 7, wherein the step (c) (S230) comprises:

(c1) (S231) converting the hyperparameter (HP1), by the hypernetwork based on the updated hypernetwork parameter (HNP2), into the second classification model parameter (MP2);
(c2) (S232) performing classification, by a classification model based on the second classification model parameter, on the verification sample for generating a second prediction label (LPD2) corresponding to the verification sample; and
(c3) (S233, S234) updating the hyperparameter (HP1, HP2) according to a second loss (L2) generated by comparing the second prediction label with a verification label (LVD) of the verification sample.

14. The machine learning method of claim 13, wherein the step (c3) comprises:
calculating the second loss by a cross-entropy calculation between the second prediction label and the verification label.

15. A non-transitory computer-readable storage medium (120) storing at least one instruction program which, when executed by a processing unit (140), causes said processing unit to perform a machine learning method (200), the machine learning method comprising:

(a) (S210) obtaining initial values of a hyperparameter and a hypernetwork parameter;
(b) (S220) generating a first classification model parameter (MP1) according to the hyperparameter (HP1) and the hypernetwork parameter (HNP1), generating a classification result by a classification model, which is configured to classify an image by detecting that the image contains one or more objects and to generate a corresponding label according to a classification result, based on the first classification model parameter relative to a training sample (TD), and updating the hypernetwork parameter (HNP1, HNP2) according to a classification result based on the first classification model parameter relative to a training sample, wherein the operation (b) executed by the processing unit comprises:

(b1) (S221) performing data augmentation, by the data augmentation model based on the hyperparameter, on the training sample for generating an augmented training sample (ETD), wherein the data augmentation model is utilized to perform a combination of one or more image processes on the training sample based on values of the hyperparameter to generate the augmented training sample;
(b2) (S222) converting the hyperparameter, by the hypernetwork based on the hypernetwork parameter, into the first classification model parameter, wherein the hypernetwork is configured to map a data point of the hyperparameter in a data augmentation space (SP1) to a data point of the first classification model parameter in a classification parameter space (SP2) based on the hypernetwork parameter, wherein the data augmentation space (SP1) is defined by axes representing different settings during data augmentation, wherein the classification parameter space (SP2) is defined by axes representing weight values of convolutional layers in the classification model;
(b3) (S223) performing classification, by the classification model based on the first classification model parameter, on the augmented training sample for generating a first prediction label (LPD1) corresponding to the augmented training sample, wherein the first classification model parameter com-

prises first weight values of convolutional layers in the classification model; and
(b4) (S224, S225) updating the hypernetwork parameter according to a first loss (L1) generated by comparing the first prediction label with a training label (LVD) of the training sample;

(c) (S230) generating a second classification model parameter (MP2) according to the hyperparameter (HP1) and the updated hypernetwork parameter (HNP2), generating another classification result by the classification model based on the second classification model parameter relative to a verification sample (VD), and updating the hyperparameter (HP1, HP2) according to the another classification result, wherein the second classification model parameter comprise second weight values of the convolutional layers in the classification model, and at least a part of the second weight values are different from the first weight values; and
(d) (S240) repeating the steps (b) and (c) for updating the hypernetwork parameter and the hyperparameter.

Patentansprüche

1. Machine-Learning-System (100), das aufweist:

eine Speichereinheit (120), die zum Speichern von Anfangswerten eines Hyperparameters (HP) und eines Hypernetzwerkparameters (HNP) konfiguriert ist;
eine Verarbeitungseinheit (140), die mit der Speichereinheit gekoppelt ist, wobei die Verarbeitungseinheit so konfiguriert ist, dass sie ein Hypernetzwerk (144), ein Klassifizierungsmodell (146), das so konfiguriert ist, dass es ein Bild durch Erfassen, dass das Bild ein oder mehrere Objekte enthält, klassifiziert und ein entsprechendes Label gemäß einem Klassifizierungsergebnis erzeugt, und ein Datenaugmentierungssmodell (142) ausführt, wobei die Verarbeitungseinheit so konfiguriert ist, dass sie Operationen ausführt zum:

(a) (S220) Erzeugen eines ersten Klassifizierungsmodellparameters (MP1) durch das Hypernetzwerk gemäß dem Hyperparameter (HP1) und dem Hypernetzwerkparameter (HNP1), Erzeugen eines Klassifizierungsergebnisses durch das Klassifizierungsmodell auf Grundlage des ersten Klassifizierungsmodellparameters relativ zu einer Trainingsprobe (TD) und Aktualisieren des Hypernetzwerkparameters (HNP1, HNP2) gemäß dem Klassifizierungsergebnis, wobei die Operation (a), die durch die Verarbeitungseinheit ausgeführt wird, aufweist:

(a1) (S221) Durchführen einer Datenaugmentierung durch das Datenaugmentierungsmodell auf Grundlage des Hyperparameters an der Trainingsprobe zum Erzeugen einer augmentierten Trainingsprobe (ETD), wobei das Datenaugmentierungsmodell verwendet wird, um eine Kombination von einer oder mehreren Bildverarbeitungen an der Trainingsprobe auf Grundlage von Werten des Hyperparameters durchzuführen, um die augmentierte Trainingsprobe zu erzeugen;
(a2) (S222) Umwandeln des Hyperparameters durch das Hypernetzwerk auf Grundlage des Hypernetzwerkparameters in den ersten Klassifizierungsmodellparameter, wobei das Hypernetzwerk konfiguriert ist, um einen Datenpunkt des Hyperparameters in einem Datenaugmentierungsraum (SP1) auf einen Datenpunkt des ersten Klassifizierungsmodellparameters in einem Klassifizierungsparameterraum (SP2) auf Grundlage des Hypernetzwerkparameters abzubilden, wobei der Datenaugmentierungsraum (SP1) durch Achsen definiert ist, die verschiedene Einstellungen während der Datenaugmentierung repräsentieren, wobei der Klassifizierungsparameterraum (SP2) durch Achsen definiert ist, die Gewichtungswerte von Faltungsschichten in dem Klassifizierungsmodell repräsentieren;
(a3) (S223) Durchführen einer Klassifizierung durch das Klassifizierungsmodell auf Grundlage des ersten Klassifizierungsmodellparameters an der augmentierten Trainingsprobe zum Erzeugen eines ersten Vorhersagelabels (LPD1), das der augmentierten Trainingsprobe entspricht, wobei der erste Klassifizierungsmodellparameter erste Gewichtungswerte von Faltungsschichten in dem Klassifizierungsmodell aufweist; und
(a4) (S224, S225) Aktualisieren des Hypernetzwerkparameters gemäß einem ersten Loss (L1), der durch Vergleichen des ersten Vorhersagelabels mit einem Trainingslabel (LVD) der Trainingsprobe erzeugt wird;

(b) (S230) Erzeugen eines zweiten Klassifizierungsmodellparameters (MP2) durch das Hypernetzwerk gemäß dem Hyperparameter (HP1) und dem aktualisierten Hypernetzwerkparameter (HNP2), Erzeugen eines weiteren Klassifizierungsergebnisses durch das Klassifizierungsmodell auf Grundlage des zweiten Klassifizierungsmodellparameters relativ zu einer Verifikationsprobe (VD) und Aktualisieren des Hyperparameters (HP1, HP2) gemäß dem weiteren Klassifizierungsergebnis, wobei der zweite Klassifizierungsmodellparameter zweite Gewichtungswerte der Faltungsschichten in dem Klassifizierungsmodell aufweist und sich zumindest ein Teil der zweiten Gewichtungswerte von den ersten Gewichtungswerten unterscheidet; und
(c) (S240) Wiederholen der Operationen (a) und (b) zum Aktualisieren des Hypernetzwerkparameters und des Hyperparameters.

2. Machine-Learning-System nach Anspruch 1, wobei die Operation (a2), die durch die Verarbeitungseinheit ausgeführt wird, aufweist:

Umwandeln des Hyperparameters durch das Hypernetzwerk auf Grundlage des Hypernetzwerkparameters und einer Vielzahl von Erkundungswerten in eine Vielzahl von Erkundungsklassifizierungsmodellparametern (MPe1, MPe2, MPe3, MPe4);
wobei die Operation (a3), die durch die Verarbeitungseinheit ausgeführt wird, aufweist:

Bilden einer Vielzahl von Erkundungsklassifizierungsmodellen (146e1, 146e2, 146e3, 146e4) durch das Klassifizierungsmodell auf Grundlage der jeweiligen Erkundungsklassifizierungsmodellparameter und Durchführen einer Klassifizierung an der augmentierten Trainingsprobe durch die jeweiligen Erkundungsklassifizierungsmodelle zum Erzeugen einer Vielzahl von ersten Vorhersagelabeln (LPD1), die der augmentierten Trainingsprobe entsprechen; und
wobei die Operation (a4), die durch die Verarbeitungseinheit ausgeführt wird, aufweist:

Berechnen einer Vielzahl von ersten Lossen (L1) durch Vergleichen der ersten Vorhersagelabels mit dem Trainingslabel der Trainingsprobe; und
Aktualisieren des Hypernetzwerkparameters (HNP1, HNP2) gemäß den Erkundungsklassifizierungsmodellen und den ersten Lossen, die den Erkundungsklassifizierungsmodellen entsprechen.

3. Machine-Learning-System nach Anspruch 2, wobei jedes der Erkundungsklassifizierungsmodelle (146e1, 146e2, 146e3, 146e4) eine Vielzahl von neuronalen Netzwerkstrukturschichten (SL1~SLn) aufweist, wobei die neuronalen Netzwerkstrukturschichten in einen ersten Strukturschichtabschnitt (P1) und einen zweiten Strukturschichtabschnitt (P2) nach dem ersten Strukturschichtabschnitt unterteilt sind, wobei jeder der Erkundungsklassifizierungsmodellparameter zum Bilden der Erkundungsklassifizierungsmodelle einen ersten Gewichtsparameterinhalt und einen zweiten Gewichtsparameterinhalt aufweist, wobei der erste Gewichtsparameterinhalt konfiguriert ist, Operationen des ersten Strukturschichtabschnitts zu bestimmen, und der zweite Gewichtsparameterinhalt konfiguriert ist, Operationen des zweiten Strukturschichtabschnitts zu bestimmen.

4. Machine-Learning-System nach Anspruch 3, wobei die zweiten Gewichtsparameterinhalte, die auf die zweiten Strukturschichtabschnitte der Erkundungsklassifizierungsmodelle angewendet werden, die gleichen sind, wobei die zweiten Strukturschichtabschnitte der Erkundungsklassifizierungsmodelle mit der gleichen Logik arbeiten.

5. Machine-Learning-System nach Anspruch 3, wobei der erste Strukturschichtabschnitt in jedem der Erkundungsklassifizierungsmodelle mindestens eine erste Faltungsschicht (SL1) aufweist, wobei die ersten Faltungsschichten unter den Erkundungsklassifizierungsmodellen voneinander verschiedene Gewichtsparameter aufweisen, wobei der zweite Strukturschichtabschnitt in jedem der Erkundungsklassifizierungsmodelle mindestens eine zweite Faltungsschicht (SL4) und mindestens eine vollständige Verbindungsschicht (SLn) aufweist, wobei die zweiten Faltungsschichten und die vollständigen Verbindungsschichten unter den Erkundungsklassifizierungsmodellen über die Erkundungsklassifizierungsmodelle hinweg gleiche Gewichtsparameter aufweisen.

6. Machine-Learning-System nach Anspruch 1, wobei die Operation (b) (S230), die durch die Verarbeitungseinheit ausgeführt wird, aufweist:

(b1) (S231) Umwandeln des Hyperparameters (HP1) durch das Hypernetzwerk auf Grundlage des aktualisierten Hypernetzwerkparameters (HNP2) in den zweiten Klassifizierungsmodellparameter (MP2);

(b2) (S232) Durchführen einer Klassifizierung durch das Klassifizierungsmodell auf Grundlage des zweiten Klassifizierungsmodellparameters an der Verifikationsprobe zum Erzeugen eines zweiten Vorhersagelabels (LPD2), das der Verifikationsprobe entspricht; und
(b3) (S233, S234) Aktualisieren des Hyperparameters (HP1, HP2) gemäß einem zweiten Loss (L2), der durch Vergleichen des zweiten Vorhersagelabels mit einem Verifikationslabel (LVD) der Verifikationsprobe erzeugt wird.

7. Machine-Learning-Verfahren (200), wobei das Machine-Learning-Verfahren durch ein Machine-Learning-System (100) durchgeführt wird, das eine Verarbeitungseinheit (140) aufweist, wobei das Machine-Learning-Verfahren aufweist:

(a) (S210) Erhalten von Anfangswerten eines Hyperparameters (HP) und eines Hypernetzwerkparameters (HNP);
(b) (S220) Erzeugen eines ersten Klassifizierungsmodellparameters (MP1) gemäß dem Hyperparameter (HP1) und dem Hypernetzwerkparameter (HNP1), Erzeugen eines Klassifizierungsergebnisses durch ein Klassifizierungsmodell, das so konfiguriert ist, dass es ein Bild durch Erfassen, dass das Bild ein oder mehrere Objekte enthält, klassifiziert und ein entsprechendes Label gemäß einem Klassifizierungsergebnis erzeugt, auf Grundlage des ersten Klassifizierungsmodellparameters relativ zu einer Trainingsprobe (TD) und Aktualisieren des Hypernetzwerkparameters (HNP1, HNP2) gemäß einem Klassifizierungsergebnis auf Grundlage des ersten Klassifizierungsmodellparameters relativ zu einer Trainingsprobe durch die Verarbeitungseinheit (140), wobei der Schritt (b) aufweist:

(b1) (S221) Durchführen einer Datenaugmentierung durch ein Datenaugmentierungsmodell auf Grundlage des Hyperparameters an der Trainingsprobe zum Erzeugen einer augmentierten Trainingsprobe (ETD), wobei das Datenaugmentierungsmodell verwendet wird, um eine Kombination von einer oder mehreren Bildverarbeitungen an der Trainingsprobe auf Grundlage von Werten des Hyperparameters durchzuführen, um die augmentierte Trainingsprobe zu erzeugen;
(b2) (S222) Umwandeln des Hyperparameters durch ein Hypernetzwerk auf Grundlage des Hypernetzwerkparameters in den ersten Klassifizierungsmodellparameter, wobei das Hypernetzwerk konfiguriert ist, um einen Datenpunkt des Hyperparameters in einem Datenaugmentierungsraum (SP1) auf einen Datenpunkt des ersten Klassifizierungsmodellparameters in einem Klassifizierungsparameterraum (SP2) auf Grundlage des Hypernetzwerkparameters abzubilden, wobei der Datenaugmentierungsraum (SP1) durch Achsen definiert ist, die verschiedene Einstellungen während der Datenaugmentierung repräsentieren, wobei der Klassifizierungsparameterraum (SP2) durch Achsen definiert ist, die Gewichtungswerte von Faltungsschichten in dem Klassifizierungsmodell repräsentieren;
(b3) (S223) Durchführen einer Klassifizierung durch das Klassifizierungsmodell auf Grundlage des ersten Klassifizierungsmodellparameters an der augmentierten Trainingsprobe zum Erzeugen eines ersten Vorhersagelabels (LPD1), das der augmentierten Trainingsprobe entspricht, wobei der erste Klassifizierungsmodellparameter erste Gewichtungswerte von Faltungsschichten in dem Klassifizierungsmodell aufweist; und
(b4) (S224, S225) Aktualisieren des Hypernetzwerkparameters gemäß einem ersten Loss (L1), der durch Vergleichen des ersten Vorhersagelabels mit einem Trainingslabel (LVD) der Trainingsprobe erzeugt wird;

(c) (S230) Erzeugen eines zweiten Klassifizierungsmodellparameters (MP2) gemäß dem Hyperparameter (HP1) und dem aktualisierten Hypernetzwerkparameter (HNP2), Erzeugen eines weiteren Klassifizierungsergebnisses durch das Klassifizierungsmodell auf Grundlage des zweiten Klassifizierungsmodellparameters relativ zu einer Verifikationsprobe (VD) und Aktualisieren des Hyperparameters (HP1, HP2) gemäß dem weiteren Klassifizierungsergebnis, wobei der zweite Klassifizierungsmodellparameter zweite Gewichtungswerte der Faltungsschichten in dem Klassifizierungsmodell aufweist und sich zumindest ein Teil der zweiten Gewichtungswerte von den ersten Gewichtungswerten unterscheidet; und
(d) (S240) Wiederholen der Schritte (b) und (c) zum Aktualisieren des Hypernetzwerkparameters und des Hyperparameters.

8. Machine-Learning-Verfahren nach Anspruch 7, wobei der Schritt (b2) aufweist:

Umwandeln des Hyperparameters durch das Hypernetzwerk auf Grundlage des Hypernetzwerkparameters und einer Vielzahl von Erkundungswerten in eine Vielzahl von

Erkundungsklassifizierungsmodellparametern;
wobei der Schritt (b3) aufweist:

Bilden einer Vielzahl von Erkundungsklassifizierungsmodellen (146e1, 146e2, 146e3, 146e4) durch das Klassifizierungsmodell auf Grundlage der jeweiligen Erkundungsklassifizierungsmodellparameter und Durchführen einer Klassifizierung an der augmentierten Trainingsprobe durch die jeweiligen Erkundungsklassifizierungsmodelle zum Erzeugen einer Vielzahl von ersten Vorhersagelabeln (LPD1), die der augmentierten Trainingsprobe entsprechen; und
wobei der Schritt (b4) aufweist:

Berechnen einer Vielzahl von ersten Lossen (L1) durch Vergleichen der ersten Vorhersagelabels mit dem Trainingslabel der Trainingsprobe; und
Aktualisieren des Hypernetzwerkparameters (HNP1, HNP2) gemäß den Erkundungsklassifizierungsmodellen und den ersten Lossen, die den Erkundungsklassifizierungsmodellen entsprechen.

9. Machine-Learning-Verfahren nach Anspruch 8, wobei in dem Schritt (b4):
Berechnen der ersten Losse durch eine Kreuzentropieberechnung zwischen den ersten Vorhersagelabeln der jeweiligen Erkundungsklassifizierungsmodelle und dem Trainingslabel.

10. Machine-Learning-Verfahren nach Anspruch 8, wobei jedes der Erkundungsklassifizierungsmodelle (146e1, 146e2, 146e3, 146e4) eine Vielzahl von neuronalen Netzwerkstrukturschichten (SL1~SLn) aufweist, wobei die neuronalen Netzwerkstrukturschichten in einen ersten Strukturschichtabschnitt (P1) und einen zweiten Strukturschichtabschnitt (P2) nach dem ersten Strukturschichtabschnitt unterteilt sind, wobei jeder der Erkundungsklassifizierungsmodellparameter zum Bilden der Erkundungsklassifizierungsmodelle einen ersten Gewichtsparameterinhalt und einen zweiten Gewichtsparameterinhalt aufweist, wobei der erste Gewichtsparameterinhalt konfiguriert ist, Operationen des ersten Strukturschichtabschnitts zu bestimmen, und der zweite Gewichtsparameterinhalt konfiguriert ist, Operationen des zweiten Strukturschichtabschnitts zu bestimmen.

11. Machine-Learning-Verfahren nach Anspruch 10, wobei die zweiten Gewichtsparameterinhalte, die auf die zweiten Strukturschichtabschnitte der Erkundungsklassifizierungsmodelle angewendet werden, die gleichen sind, wobei die zweiten Strukturschichtabschnitte der Erkundungsklassifizierungsmodelle mit der gleichen Logik arbeiten.

12. Machine-Learning-Verfahren nach Anspruch 10, wobei der erste Strukturschichtabschnitt in jedem der Erkundungsklassifizierungsmodelle mindestens eine erste Faltungsschicht (SL1) aufweist, wobei die ersten Faltungsschichten unter den Erkundungsklassifizierungsmodellen voneinander verschiedene Gewichtsparameter aufweisen, wobei der zweite Strukturschichtabschnitt in jedem der Erkundungsklassifizierungsmodelle mindestens eine zweite Faltungsschicht (SL4) und mindestens eine vollständige Verbindungsschicht (SLn) aufweist, wobei die zweiten Faltungsschichten und die vollständigen Verbindungsschichten unter den Erkundungsklassifizierungsmodellen über die Erkundungsklassifizierungsmodelle hinweg gleiche Gewichtsparameter aufweisen.

13. Machine-Learning-Verfahren nach Anspruch 7, wobei der Schritt (c) (S230) aufweist:

(c1) (S231) Umwandeln des Hyperparameters (HP1) durch das Hypernetzwerk auf Grundlage des aktualisierten Hypernetzwerkparameters (HNP2) in den zweiten Klassifizierungsmodellparameter (MP2);
(c2) (S232) Durchführen einer Klassifizierung durch ein Klassifizierungsmodell auf Grundlage des zweiten Klassifizierungsmodellparameters an der Verifikationsprobe zum Erzeugen eines zweiten Vorhersagelabels (LPD2), das der Verifikationsprobe entspricht; und
(c3) (S233, S234) Aktualisieren des Hyperparameters (HP1, HP2) gemäß einem zweiten Loss (L2), der durch Vergleichen des zweiten Vorhersagelabels mit einem Verifikationslabel (LVD) der Verifikationsprobe erzeugt wird.

14. Machine-Learning-Verfahren nach Anspruch 13, wobei der Schritt (c3) aufweist:
Berechnen des zweiten Losses durch eine Kreuzentropieberechnung zwischen dem zweiten Vorhersagelabel und dem Verifikationslabel.

15. Nicht-transitorisches computerlesbares Speichermedium (120), das mindestens ein Anweisungsprogramm speichert, das, wenn es durch eine Verarbeitungseinheit (140) ausgeführt wird, die Verarbeitungseinheit veranlasst, ein Machine-Learning-Verfahren (200) durchzuführen, wobei das Machine-Learning-Verfahren aufweist:

(a) (S210) Erhalten von Anfangswerten eines Hyperparameters und eines Hypernetzwerkparameters;

(b) (S220) Erzeugen eines ersten Klassifizierungsmodellparameters (MP1) gemäß dem Hyperparameter (HP1) und dem Hypernetzwerkparameter (HNP1), Erzeugen eines Klassifizierungsergebnisses durch ein Klassifizierungsmodell, das so konfiguriert ist, dass es ein Bild durch Erfassen, dass das Bild ein oder mehrere Objekte enthält, klassifiziert und ein entsprechendes Label gemäß einem Klassifizierungsergebnis erzeugt, auf Grundlage des ersten Klassifizierungsmodellparameters relativ zu einer Trainingsprobe (TD) und Aktualisieren des Hypernetzwerkparameters (HNP1, HNP2) gemäß einem Klassifizierungsergebnis auf Grundlage des ersten Klassifizierungsmodellparameters relativ zu einer Trainingsprobe, wobei die Operation (b), die durch die Verarbeitungseinheit ausgeführt wird, aufweist:

(b1) (S221) Durchführen einer Datenaugmentierung durch das Datenaugmentierungsmodell auf Grundlage des Hyperparameters an der Trainingsprobe zum Erzeugen einer augmentierten Trainingsprobe (ETD), wobei das Datenaugmentierungsmodell verwendet wird, um eine Kombination von einer oder mehreren Bildverarbeitungen an der Trainingsprobe auf Grundlage von Werten des Hyperparameters durchzuführen, um die augmentierte Trainingsprobe zu erzeugen;
(b2) (S222) Umwandeln des Hyperparameters durch das Hypernetzwerk auf Grundlage des Hypernetzwerkparameters in den ersten Klassifizierungsmodellparameter, wobei das Hypernetzwerk konfiguriert ist, um einen Datenpunkt des Hyperparameters in einem Datenaugmentierungsraum (SP1) auf einen Datenpunkt des ersten Klassifizierungsmodellparameters in einem Klassifizierungsparameterraum (SP2) auf Grundlage des Hypernetzwerkparameters abzubilden, wobei der Datenaugmentierungsraum (SP1) durch Achsen definiert ist, die verschiedene Einstellungen während der Datenaugmentierung repräsentieren, wobei der Klassifizierungsparameterraum (SP2) durch Achsen definiert ist, die Gewichtungswerte von Faltungsschichten in dem Klassifizierungsmodell repräsentieren;
(b3) (S223) Durchführen einer Klassifizierung durch das Klassifizierungsmodell auf Grundlage des ersten Klassifizierungsmodellparameters an der augmentierten Trainingsprobe zum Erzeugen eines ersten Vorhersagelabels (LPD1), das der augmentierten Trainingsprobe entspricht, wobei der erste Klassifizierungsmodellparameter erste Gewichtungswerte von Faltungsschichten in dem Klassifizierungsmodell aufweist; und
(b4) (S224, S225) Aktualisieren des Hypernetzwerkparameters gemäß einem ersten Loss (L1), der durch Vergleichen des ersten Vorhersagelabels mit einem Trainingslabel (LVD) der Trainingsprobe erzeugt wird;

(c) (S230) Erzeugen eines zweiten Klassifizierungsmodellparameters (MP2) gemäß dem Hyperparameter (HP1) und dem aktualisierten Hypernetzwerkparameter (HNP2), Erzeugen eines weiteren Klassifizierungsergebnisses durch das Klassifizierungsmodell auf Grundlage des zweiten Klassifizierungsmodellparameters relativ zu einer Verifikationsprobe (VD) und Aktualisieren des Hyperparameters (HP1, HP2) gemäß dem weiteren Klassifizierungsergebnis, wobei der zweite Klassifizierungsmodellparameter zweite Gewichtungswerte der Faltungsschichten in dem Klassifizierungsmodell aufweist und sich zumindest ein Teil der zweiten Gewichtungswerte von den ersten Gewichtungswerten unterscheidet; und
(d) (S240) Wiederholen der Schritte (b) und (c) zum Aktualisieren des Hypernetzwerkparameters und des Hyperparameters.

Revendications

1. Système d'apprentissage par machine (100), comprenant :

une unité de mémoire (120) configurée pour stocker des valeurs initiales d'un hyperparamètre (HP) et d'un paramètre d'hyperréseau (HNP) ;
une unité de traitement (140), couplée à l'unité de mémoire, dans lequel l'unité de traitement est configurée pour exécuter un hyperréseau (144), un modèle de classification (146) qui est configuré pour classer une image en détectant que l'image contient un ou plusieurs objets et pour générer une étiquette correspondante selon un résultat de classification, et un modèle d'augmentation de données (142), l'unité de traitement est configurée pour exécuter des opérations consistant à :

(a) (S220) générer un premier paramètre de modèle de classification (MP1) au moyen de l'hyperréseau selon l'hyperparamètre (HP1) et le paramètre d'hyperréseau (HNP1), générer un résultat de classification au moyen du modèle de classification

sur la base du premier paramètre de modèle de classification par rapport à un échantillon d'apprentissage (TD), et mettre à jour le paramètre d'hyperréseau (HNP1, HNP2) selon le résultat de classification, dans lequel l'opération (a) exécutée par l'unité de traitement comprend :

(a1) (S221) la réalisation d'une augmentation de données, au moyen du modèle d'augmentation de données sur la base de l'hyperparamètre, sur l'échantillon d'apprentissage pour générer un échantillon d'apprentissage augmenté (ETD), dans lequel le modèle d'augmentation de données est utilisé pour effectuer une combinaison d'un ou de plusieurs processus d'image sur l'échantillon d'apprentissage sur la base de valeurs de l'hyperparamètre pour générer l'échantillon d'apprentissage augmenté ;
(a2) (S222) la conversion de l'hyperparamètre, au moyen de l'hyperréseau sur la base du paramètre d'hyperréseau, en premier paramètre de modèle de classification, dans lequel l'hyperréseau est configuré pour mapper un point de données de l'hyperparamètre dans un espace d'augmentation de données (SP1) par rapport à un point de données du premier paramètre de modèle de classification dans un espace de paramètre de classification (SP2) sur la base du paramètre d'hyperréseau, dans lequel l'espace d'augmentation de données (SP1) est défini par des axes représentant différents réglages pendant une augmentation de données, dans lequel l'espace de paramètre de classification (SP2) est défini par des axes représentant des valeurs de poids de couches convolutionnelles dans le modèle de classification ;
(a3) (S223) la réalisation d'une classification, au moyen du modèle de classification sur la base du premier paramètre de modèle de classification, sur l'échantillon d'apprentissage augmenté pour générer une première étiquette de prédiction (LPD1) correspondant à l'échantillon d'apprentissage augmenté, dans lequel le premier paramètre de modèle de classification comprend des premières valeurs de poids de couches convolutionnelles dans le modèle de classification ; et
(a4) (S224, S225) la mise à jour du paramètre d'hyperréseau selon une première perte (L1) générée en comparant la première étiquette de prédiction avec une étiquette d'apprentissage (LVD) de l'échantillon d'apprentissage ;

(b) (S230) générer un second paramètre de modèle de classification (MP2) au moyen de l'hyperréseau selon l'hyperparamètre (HP1) et le paramètre d'hyperréseau mis à jour (HNP2), générer un autre résultat de classification au moyen du modèle de classification sur la base du second paramètre de modèle de classification par rapport à un échantillon de vérification (VD), et mettre à jour l'hyperparamètre (HP1, HP2) selon l'autre résultat de classification, dans lequel le second paramètre de modèle de classification comprend des secondes valeurs de poids des couches convolutionnelles dans le modèle de classification, et au moins une partie des secondes valeurs de poids sont différentes des premières valeurs de poids ; et
(c) (S240) répéter les opérations (a) et (b) pour mettre à jour le paramètre d'hyperréseau et l'hyperparamètre.

2. Système d'apprentissage par machine selon la revendication 1, dans lequel l'opération (a2) exécutée par l'unité de traitement comprend :

la conversion de l'hyperparamètre, au moyen de l'hyperréseau sur la base du paramètre d'hyperréseau et d'une pluralité de valeurs d'exploration, en une pluralité de paramètres de modèle de classification d'exploration (MPe1, MPe2, MPe3, MPe4) ;
dans lequel l'opération (a3) exécutée par l'unité de traitement comprend :

la formation d'une pluralité de modèles de classification d'exploration (146e1, 146e2, 146e3, 146e4) au moyen du modèle de classification sur la base des paramètres de modèle de classification d'exploration, respectivement, et la réalisation d'une classification sur l'échantillon d'apprentissage augmenté au moyen des modèles de classification d'exploration, respectivement, pour générer une pluralité de premières étiquettes de prédiction (LPD1) correspondant à l'échantillon d'apprentissage augmenté ; et
dans lequel l'opération (a4) exécutée par l'unité de traitement comprend :

le calcul d'une pluralité de premières

pertes (L1) en comparant les premières étiquettes de prédiction à l'étiquette d'apprentissage de l'échantillon d'apprentissage ; et
la mise à jour du paramètre d'hyperréseau (HNP1, HNP2) selon les modèles de classification d'exploration et les premières pertes correspondant aux modèles de classification d'exploration.

3. Système d'apprentissage par machine selon la revendication 2, dans lequel chacun des modèles de classification d'exploration (146e1, 146e2, 146e3, 146e4) comprend une pluralité de couches structurelles de réseau neuronal (SL1~SLn), les couches structurelles du réseau neuronal sont divisées en une première partie de couche structurelle (P1) et une seconde partie de couche structurelle (P2) après la première partie de couche structurelle, chacun des paramètres de modèle de classification d'exploration pour former les modèles de classification d'exploration comprend un premier contenu de paramètre de poids et un second contenu de paramètre de poids, le premier contenu de paramètre de poids est configuré pour déterminer des opérations de la première partie de couche structurelle et le second contenu de paramètre de poids est configuré pour déterminer des opérations de la seconde partie de couche structurelle.

4. Système d'apprentissage par machine selon la revendication 3, dans lequel les seconds contenus de paramètre de poids appliqués aux secondes parties de couche structurelle des modèles de classification d'exploration sont identiques, les secondes parties de couche structurelle des modèles de classification d'exploration fonctionnent avec la même logique.

5. Système d'apprentissage par machine selon la revendication 3, dans lequel la première partie de couche structurelle dans chacun des modèles de classification d'exploration comprend au moins une première couche convolutionnelle (SL1), les premières couches convolutionnelles parmi les modèles de classification d'exploration présentent des paramètres de poids différents les uns des autres, la seconde partie de couche structurelle dans chacun des modèles de classification d'exploration comprend au moins une seconde couche convolutionnelle (SL4) et au moins une couche de connexion complète (SLn), les secondes couches convolutionnelles et les couches de connexion complète parmi les modèles de classification d'exploration présentent des paramètres de poids identiques à travers les modèles de classification d'exploration.

6. Système d'apprentissage par machine selon la revendication 1, dans lequel l'opération (b) (S230) exécutée par l'unité de traitement comprend :

(b1) (S231) la conversion de l'hyperparamètre (HP1), au moyen de l'hyperréseau sur la base du paramètre d'hyperréseau mis à jour (HNP2), en second paramètre de modèle de classification (MP2) ;
(b2) (S232) la réalisation d'une classification, au moyen du modèle de classification sur la base du second paramètre de modèle de classification, sur l'échantillon de vérification pour générer une seconde étiquette de prédiction (LPD2) correspondant à l'échantillon de vérification ; et
(b3) (S233, S234) la mise à jour de l'hyperparamètre (HP1, HP2) selon une seconde perte (L2) générée en comparant la seconde étiquette de prédiction à une étiquette de vérification (LVD) de l'échantillon de vérification.

7. Procédé d'apprentissage par machine (200), dans lequel le procédé d'apprentissage par machine est exécuté par un système d'apprentissage par machine (100) comprenant une unité de traitement (140), le procédé d'apprentissage par machine comprenant :

(a) (S210) l'obtention de valeurs initiales d'un hyperparamètre (HP) et d'un paramètre d'hyperréseau (HNP) ;
(b) (S220) la génération d'un premier paramètre de modèle de classification (MP1) selon l'hyperparamètre (HP1) et le paramètre d'hyperréseau (HNP1), la génération d'un résultat de classification au moyen d'un modèle de classification qui est configuré pour classer une image en détectant que l'image contient un ou plusieurs objets et pour générer une étiquette correspondante selon un résultat de classification, sur la base du premier paramètre de modèle de classification par rapport à un échantillon d'apprentissage (TD), et la mise à jour du paramètre d'hyperréseau (HNP1, HNP2) selon un résultat de classification sur la base du premier paramètre de modèle de classification par rapport à un échantillon d'apprentissage au moyen de l'unité de traitement (140), dans lequel l'étape (b) comprend :

(b1) (S221) la réalisation d'une augmentation de données, au moyen d'un modèle d'augmentation de données sur la base de l'hyperparamètre, sur l'échantillon d'apprentissage pour générer un échantillon d'apprentissage augmenté (ETD), dans lequel le modèle d'augmentation de données est utilisé pour effectuer une combinaison d'un ou de plusieurs processus d'image sur

l'échantillon d'apprentissage sur la base de valeurs de l'hyperparamètre pour générer l'échantillon d'apprentissage augmenté b

(b2) (S222) la conversion de l'hyperparamètre, au moyen d'un hyperréseau sur la base du paramètre d'hyperréseau, en premier paramètre de modèle de classification, dans lequel l'hyperréseau est configuré pour mapper un point de données de l'hyperparamètre dans un espace d'augmentation de données (SP1) par rapport à un point de données du premier paramètre de modèle de classification dans un espace de paramètre de classification (SP2) sur la base du paramètre d'hyperréseau, dans lequel l'espace d'augmentation de données (SP1) est défini par des axes représentant différents réglages pendant une augmentation de données, dans lequel l'espace de paramètre de classification (SP2) est défini par des axes représentant des valeurs de poids de couches convolutionnelles dans le modèle de classification ;

(b3) (S223) la réalisation d'une classification, au moyen du modèle de classification sur la base du premier paramètre de modèle de classification, sur l'échantillon d'apprentissage augmenté pour générer une première étiquette de prédiction (LPD1) correspondant à l'échantillon d'apprentissage augmenté, dans lequel le premier paramètre de modèle de classification comprend des premières valeurs de poids de couches convolutionnelles dans le modèle de classification ; et

(b4) (S224, S225) la mise à jour du paramètre d'hyperréseau selon une première perte (L1) générée en comparant la première étiquette de prédiction à une étiquette d'apprentissage (LVD) de l'échantillon d'apprentissage ;

(c) (S230) la génération d'un second paramètre de modèle de classification (MP2) selon l'hyperparamètre (HP1) et le paramètre d'hyperréseau mis à jour (HNP2), la génération d'un autre résultat de classification au moyen du modèle de classification sur la base du second paramètre de modèle de classification par rapport à un échantillon de vérification (VD), et la mise à jour de l'hyperparamètre (HP1, HP2) selon l'autre résultat de classification, dans lequel le second paramètre de modèle de classification comprend des secondes valeurs de poids des couches convolutionnelles dans le modèle de classification, et au moins une partie des secondes valeurs de poids sont différentes des premières valeurs de poids ; et

(d) (S240) la répétition des étapes (b) et (c) pour mettre à jour le paramètre d'hyperréseau et l'hyperparamètre.

8. Procédé d'apprentissage par machine selon la revendication 7, dans lequel l'étape (b2) comprend :

la conversion de l'hyperparamètre, au moyen de l'hyperréseau sur la base du paramètre d'hyperréseau et d'une pluralité de valeurs d'exploration, en une pluralité de paramètres de modèle de classification d'exploration ;

dans lequel l'étape (b3) comprend :

la formation d'une pluralité de modèles de classification d'exploration (146e1, 146e2, 146e3, 146e4) au moyen du modèle de classification sur la base des paramètres de modèle de classification d'exploration, respectivement, et la réalisation d'une classification sur l'échantillon d'apprentissage augmenté au moyen des modèles de classification d'exploration, respectivement, pour générer une pluralité de premières étiquettes de prédiction (LPD1) correspondant à l'échantillon d'apprentissage augmenté ; et

dans lequel l'étape (b4) comprend :

le calcul d'une pluralité de premières pertes (L1) en comparant les premières étiquettes de prédiction à l'étiquette d'apprentissage de l'échantillon d'apprentissage ; et

la mise à jour du paramètre d'hyperréseau (HNP1, HNP2) selon les modèles de classification d'exploration et les premières pertes correspondant aux modèles de classification d'exploration.

9. Procédé d'apprentissage par machine selon la revendication 8, dans lequel, dans l'étape (b4) : le calcul des premières pertes par un calcul d'entropie croisée entre les premières étiquettes de prédiction des modèles de classification d'exploration et l'étiquette d'apprentissage, respectivement.

10. Procédé d'apprentissage par machine selon la revendication 8, dans lequel chacun des modèles de classification d'exploration (146e1, 146e2, 146e3, 146e4) comprend une pluralité de couches structurelles de réseau neuronal (SL1~SLn), les couches structurelles du réseau neuronal sont divisées en une première partie de couche structurelle (P1) et une seconde partie de couche structurelle (P2) après la première partie de couche structurelle, chacun des paramètres de modèle de classification d'explora-

tion pour former les modèles de classification d'exploration comprend un premier contenu de paramètre de poids et un second contenu de paramètre de poids, le premier contenu de paramètre de poids est configuré pour déterminer des opérations de la première partie de couche structurelle, et le second contenu de paramètre de poids est configuré pour déterminer des opérations de la seconde partie de couche structurelle.

11. Procédé d'apprentissage par machine selon la revendication 10, dans lequel les seconds contenus de paramètre de poids appliqués aux secondes parties de couche structurelle des modèles de classification d'exploration sont identiques, les secondes parties de couche structurelle des modèles de classification d'exploration fonctionnent avec la même logique.

12. Procédé d'apprentissage par machine selon la revendication 10, dans lequel la première partie de couche structurelle dans chacun des modèles de classification d'exploration comprend au moins une première couche convolutionnelle (SL1), les premières couches convolutionnelles parmi les modèles de classification d'exploration présentent des paramètres de poids différents les uns des autres, la seconde partie de couche structurelle dans chacun des modèles de classification d'exploration comprend au moins une seconde couche convolutionnelle (SL4) et au moins une couche de connexion complète (SLn), les secondes couches convolutionnelles et les couches de connexion complète parmi les modèles de classification d'exploration présentent des paramètres de poids identiques à travers les modèles de classification d'exploration.

13. Procédé d'apprentissage par machine selon la revendication 7, dans lequel l'étape (c) (S230) comprend :

(c1) (S231) la conversion de l'hyperparamètre (HP1), au moyen de l'hyperréseau sur la base du paramètre d'hyperréseau mis à jour (HNP2), en second paramètre de modèle de classification (MP2) ;
(c2) (S232) la réalisation d'une classification, au moyen d'un modèle de classification sur la base du second paramètre de modèle de classification, sur l'échantillon de vérification pour générer une seconde étiquette de prédiction (LPD2) correspondant à l'échantillon de vérification ; et
(c3) (S233, S234) la mise à jour de l'hyperparamètre (HP1, HP2) selon une seconde perte (L2) générée en comparant la seconde étiquette de prédiction à une étiquette de vérification (LVD) de l'échantillon de vérification.

14. Procédé d'apprentissage par machine selon la revendication 13, dans lequel l'étape (c3) comprend : le calcul de la seconde perte au moyen d'un calcul d'entropie croisée entre la seconde étiquette de prédiction et l'étiquette de vérification.

15. Support de stockage non transitoire lisible par ordinateur (120) stockant au moins un programme d'instructions qui, lorsqu'il est exécuté par une unité de traitement (140), amène ladite unité de passage à réaliser un procédé d'apprentissage par machine (200), le procédé d'apprentissage par machine par comprenant :

(a) (S210) l'obtention de valeurs initiales d'un hyperparamètre et d'un paramètre d'hyperréseau ;
(b) (S220) la génération d'un premier paramètre de modèle de classification (MP1) selon l'hyperparamètre (HP1) et le paramètre d'hyperréseau (HNP1), la génération d'un résultat de classification au moyen d'un modèle de classification, qui est configuré pour classer une image en détectant que l'image contient un ou plusieurs objets et pour générer une étiquette correspondante selon un résultat de classification, sur la base du premier paramètre de modèle de classification par rapport à un échantillon d'apprentissage (TD), et la mise à jour du paramètre d'hyperréseau (HNP1, HNP2) selon un résultat de classification sur la base du premier paramètre de modèle de classification par rapport à un échantillon d'apprentissage, dans lequel l'opération (b) exécutée par l'unité de traitement comprend :

(b1) (S221) la réalisation d'une augmentation de données, au moyen du modèle d'augmentation de données sur la base de l'hyperparamètre, sur l'échantillon d'apprentissage pour générer un échantillon d'apprentissage augmenté (ETD), dans lequel le modèle d'augmentation de données est utilisé pour effectuer une combinaison d'un ou de plusieurs processus d'image sur l'échantillon d'apprentissage sur la base de valeurs de l'hyperparamètre pour générer l'échantillon d'apprentissage augmenté ;
(b2) (S222) la conversion de l'hyperparamètre, au moyen de l'hyperréseau sur la base du paramètre d'hyperréseau, en premier paramètre de modèle de classification, dans lequel l'hyperréseau est configuré pour mapper un point de données de l'hyperparamètre dans un espace d'augmentation de données (SP1) par rapport à un point de données du premier paramètre de modèle de classification dans un espace de

paramètre de classification (SP2) sur la base du paramètre d'hyperréseau, dans lequel l'espace d'augmentation de données (SP1) est défini par des axes représentant différents réglages pendant une augmentation de données, dans lequel l'espace de paramètre de classification (SP2) est défini par des axes représentant des valeurs de poids de couches convolutionnelles dans le modèle de classification ;

(b3) (S223) la réalisation d'une classification, au moyen du modèle de classification sur la base du premier paramètre de modèle de classification, sur l'échantillon d'apprentissage augmenté pour générer une première étiquette de prédiction (LPD1) correspondant à l'échantillon d'apprentissage augmenté, dans lequel le premier paramètre de modèle de classification comprend des premières valeurs de poids de couches convolutionnelles dans le modèle de classification ; et

(b4) (S224, S225) la mise à jour du paramètre d'hyperréseau selon une première perte (L1) générée en comparant la première étiquette de prédiction à une étiquette d'apprentissage (LVD) de l'échantillon d'apprentissage ;

(c) (S230) la génération d'un second paramètre de modèle de classification (MP2) selon l'hyperparamètre (HP1) et le paramètre d'hyperréseau mis à jour (HNP2), la génération d'un autre résultat de classification au moyen du modèle de classification sur la base du second paramètre de modèle de classification par rapport à un échantillon de vérification (VD), et la mise à jour de l'hyperparamètre (HP1, HP2) selon l'autre résultat de classification, dans lequel le second paramètre de modèle de classification comprend des secondes valeurs de poids des couches convolutionnelles dans le modèle de classification, et au moins une partie des secondes valeurs de poids sont différentes des premières valeurs de poids ; et

(d) (S240) la répétition des étapes (b) et (c) pour mettre à jour le paramètre d'hyperréseau et l'hyperparamètre.

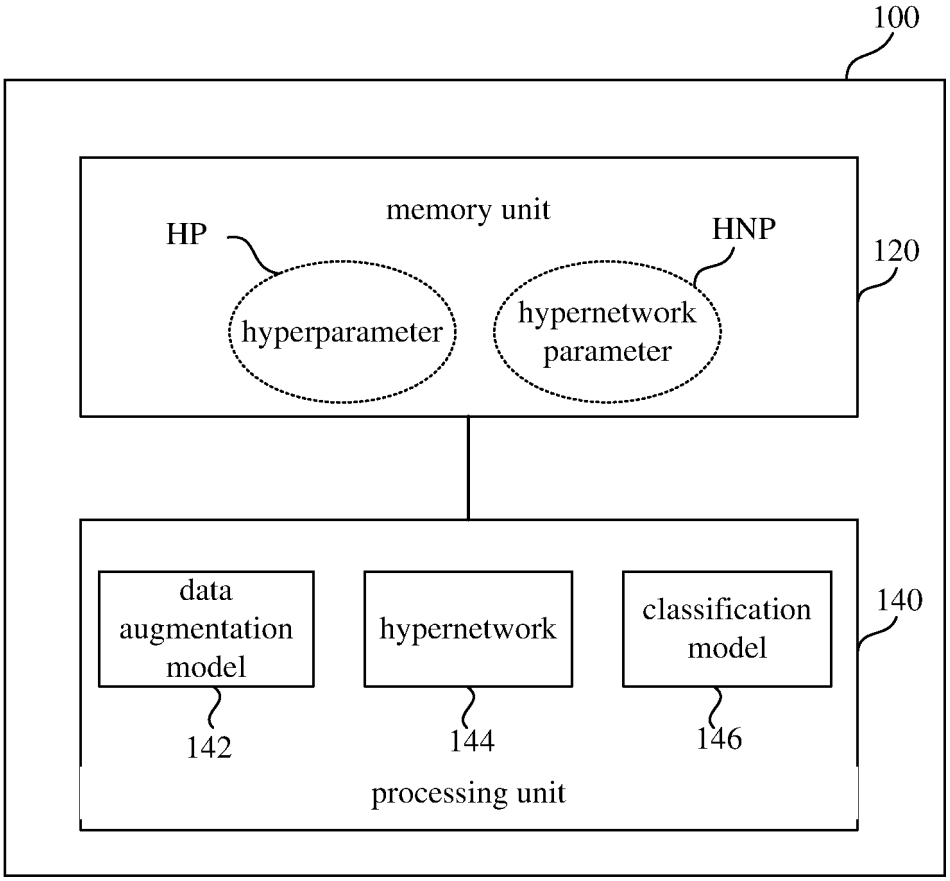

100
memory unit
HP
HNP
120
hyperparameter
hypernetwork parameter
data augmentation model
hypernetwork
classification model
140
142
144
146
processing unit

FIG. 1

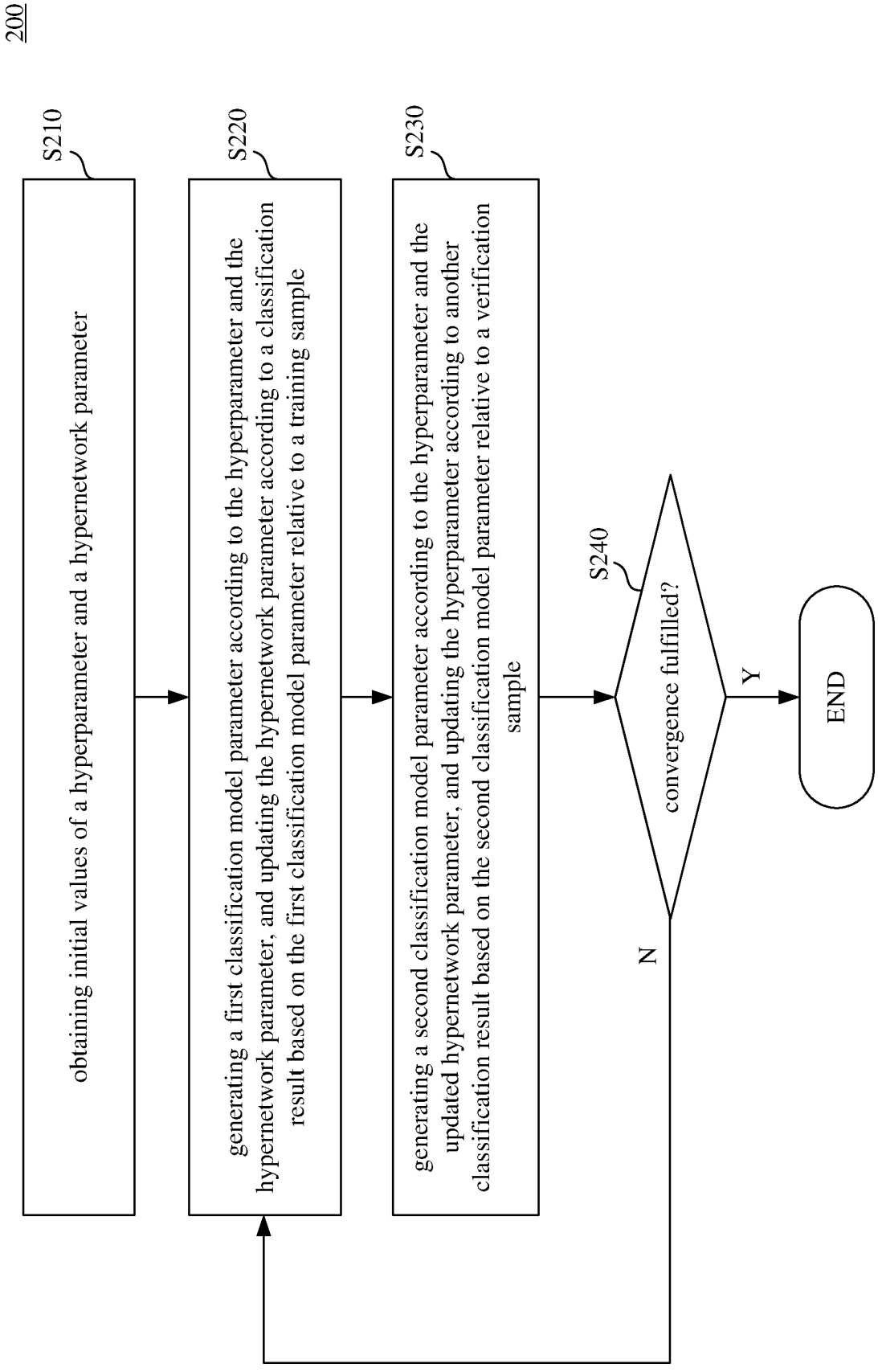

200
S210
obtaining initial values of a hyperparameter and a hypernetwork parameter
S220
generating a first classification model parameter according to the hyperparameter and the hypernetwork parameter, and updating the hypernetwork parameter according to a classification result based on the first classification model parameter relative to a training sample
S230
generating a second classification model parameter according to the hyperparameter and the updated hypernetwork parameter, and updating the hyperparameter according to another classification result based on the second classification model parameter relative to a verification sample
S240
convergence fulfilled?
N
Y
END

FIG. 2

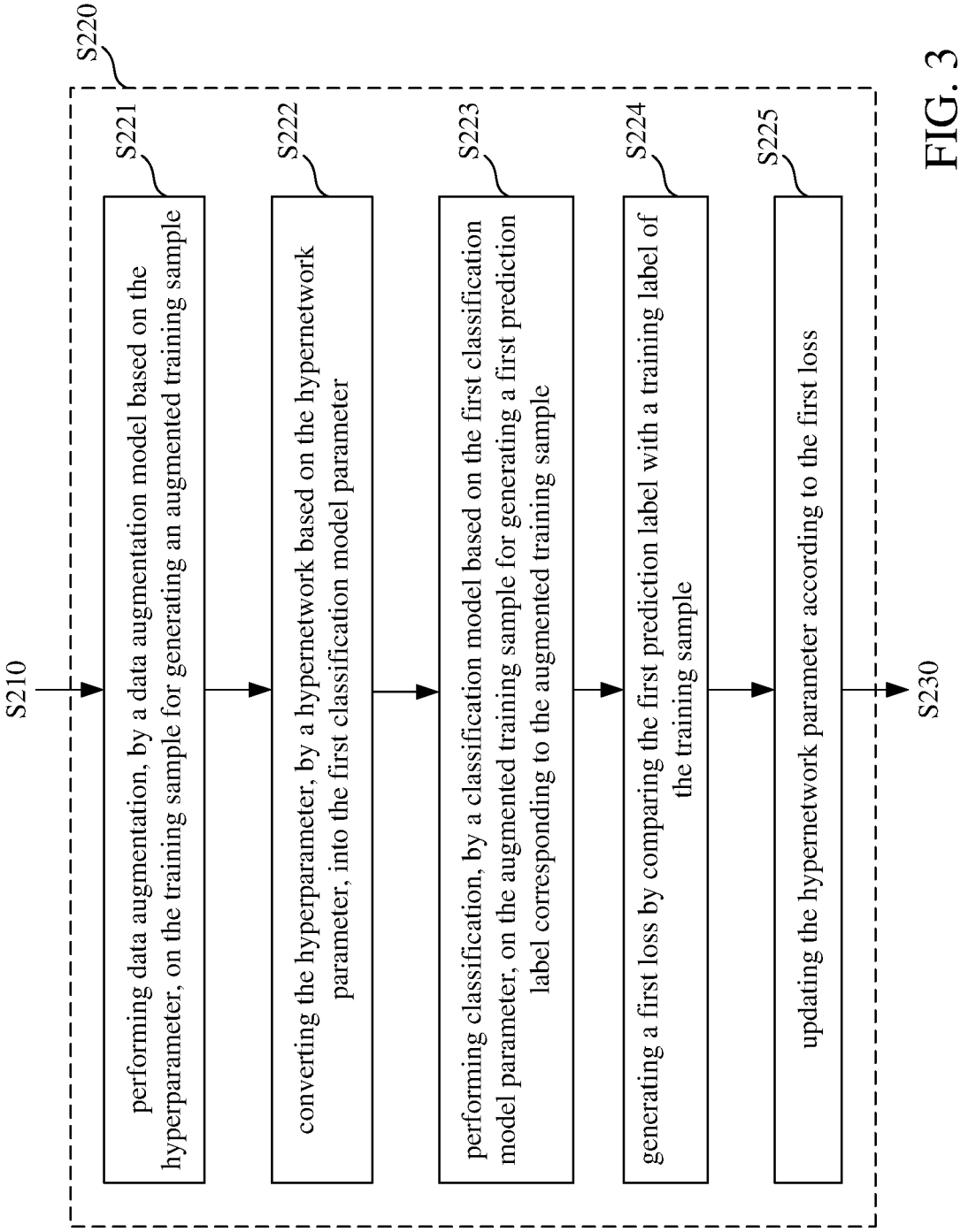

S210
S220
S221
performing data augmentation, by a data augmentation model based on the hyperparameter, on the training sample for generating an augmented training sample
S222
converting the hyperparameter, by a hypernetwork based on the hypernetwork parameter, into the first classification model parameter
S223
performing classification, by a classification model based on the first classification model parameter, on the augmented training sample for generating a first prediction label corresponding to the augmented training sample
S224
generating a first loss by comparing the first prediction label with a training label of the training sample
S225
updating the hypernetwork parameter according to the first loss
S230

FIG. 3

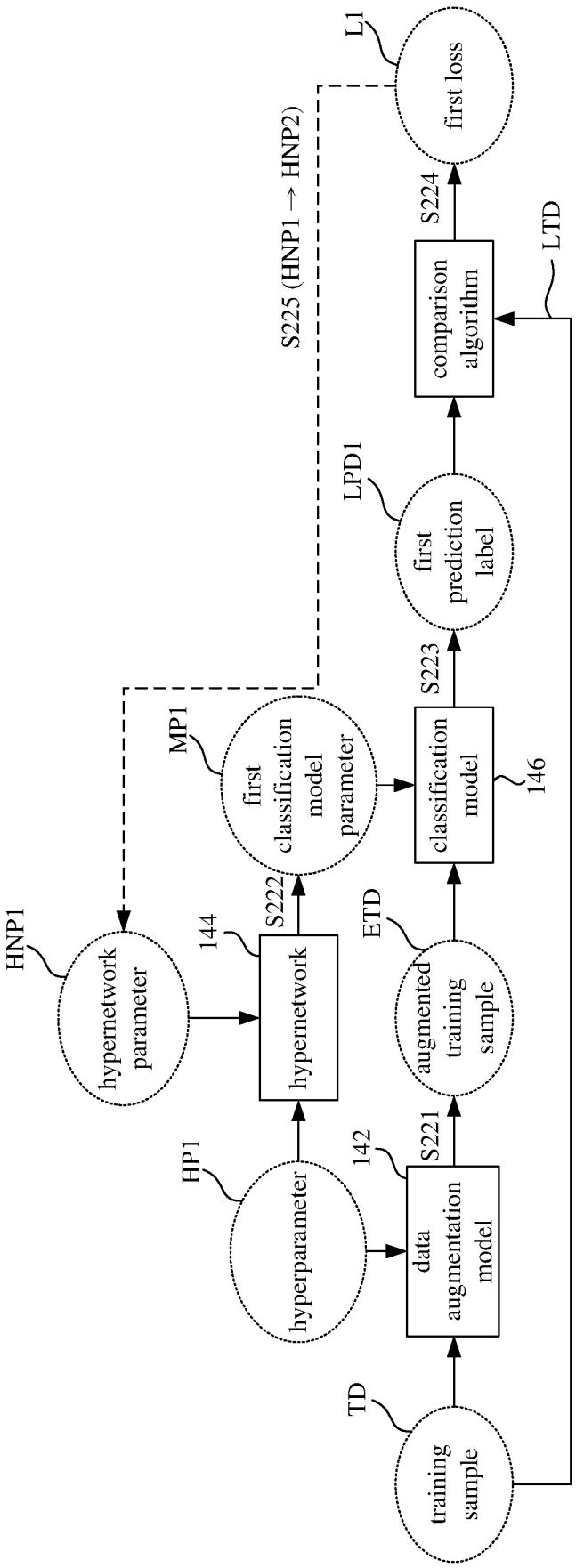

training sample
TD
data augmentation model
142
S221
hyperparameter
HP1
hypernetwork
144
S222
hypernetwork parameter
HNP1
augmented training sample
ETD
classification model
146
S223
first classification model parameter
MP1
first prediction label
LPD1
comparison algorithm
S224
first loss
L1
LTD
S225 (HNP1 → HNP2)

FIG. 4

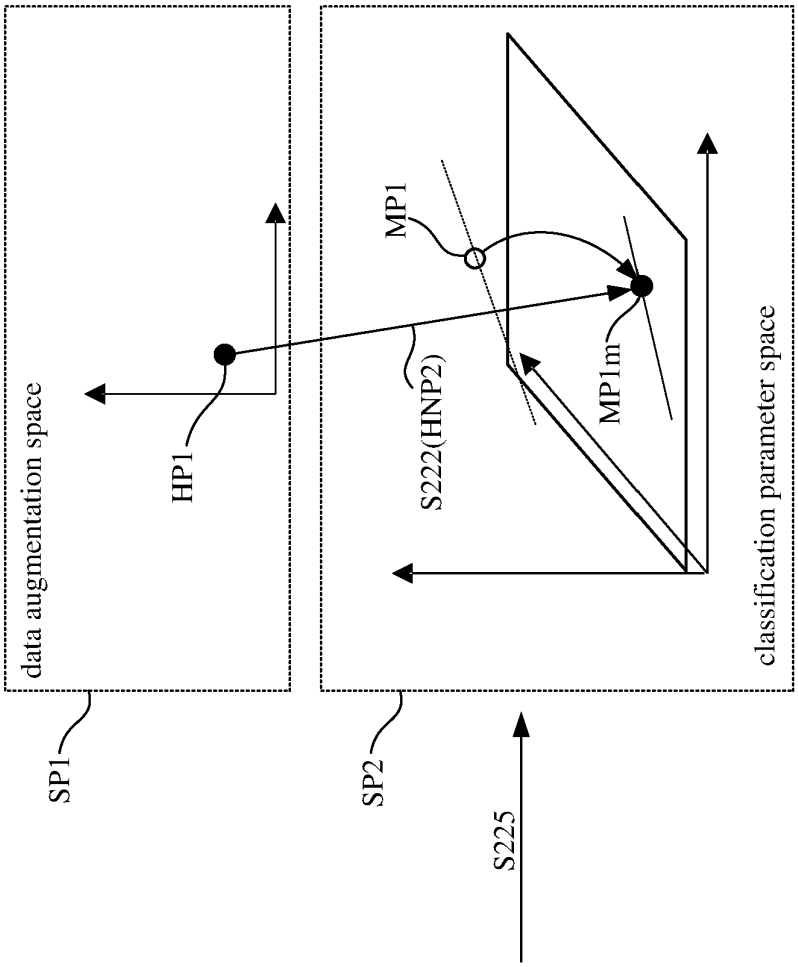

data augmentation space
SP1
HP1
SP2
S222(HNP2)
MP1
MP1m
classification parameter space
S225

FIG. 5B

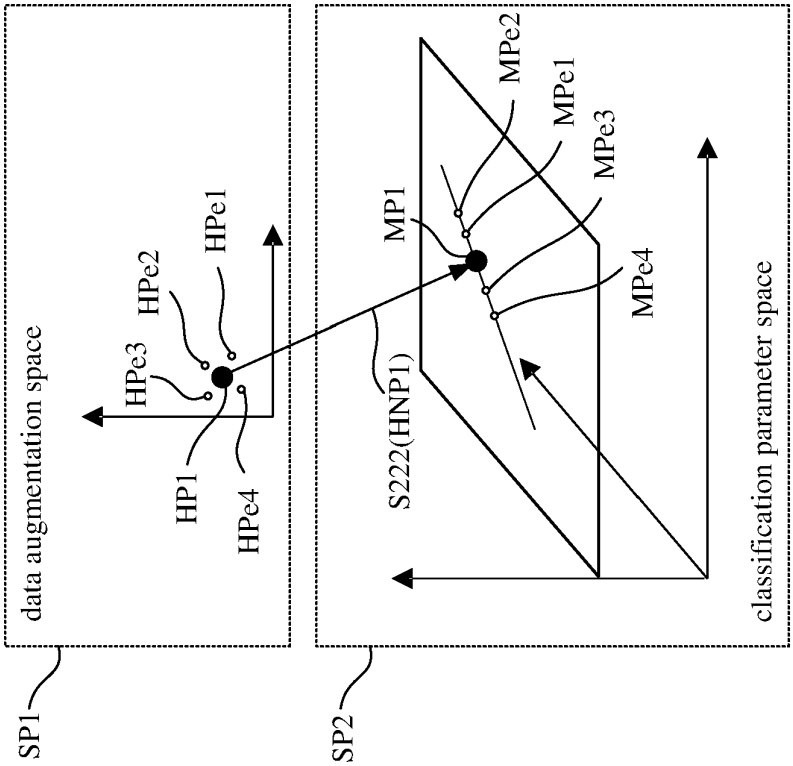
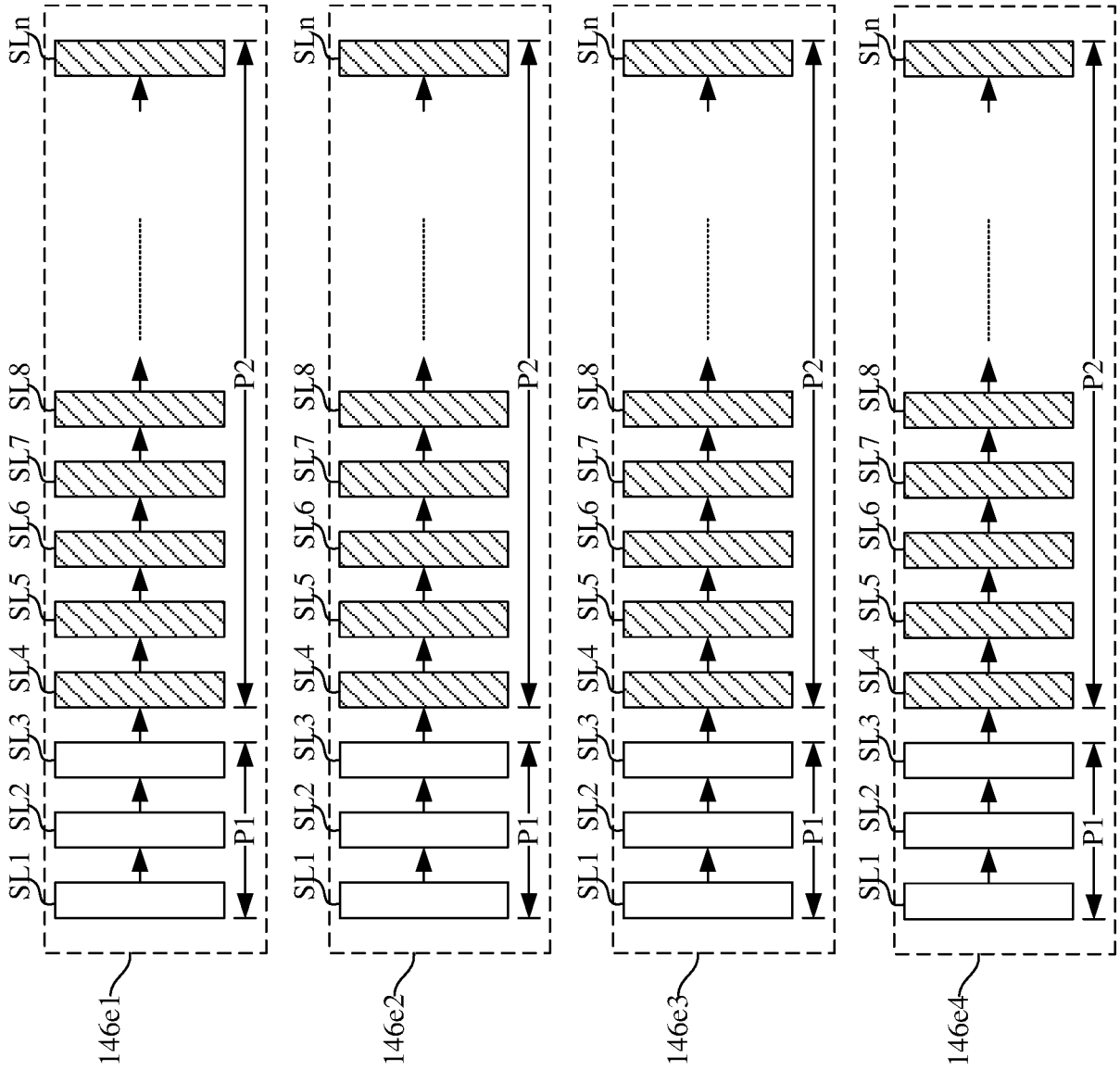

data augmentation space
SP1
HPe3
HPe2
HPe1
HP1
HPe4
SP2
S222(HNP1)
MP1
MPe2
MPe1
MPe3
MPe4
classification parameter space

FIG. 5A

FIG. 6


SL1
SL2
SL3
SL4
SL5
SL6
SL7
SL8
SLn
P1
P2
146e1
146e2
146e3
146e4

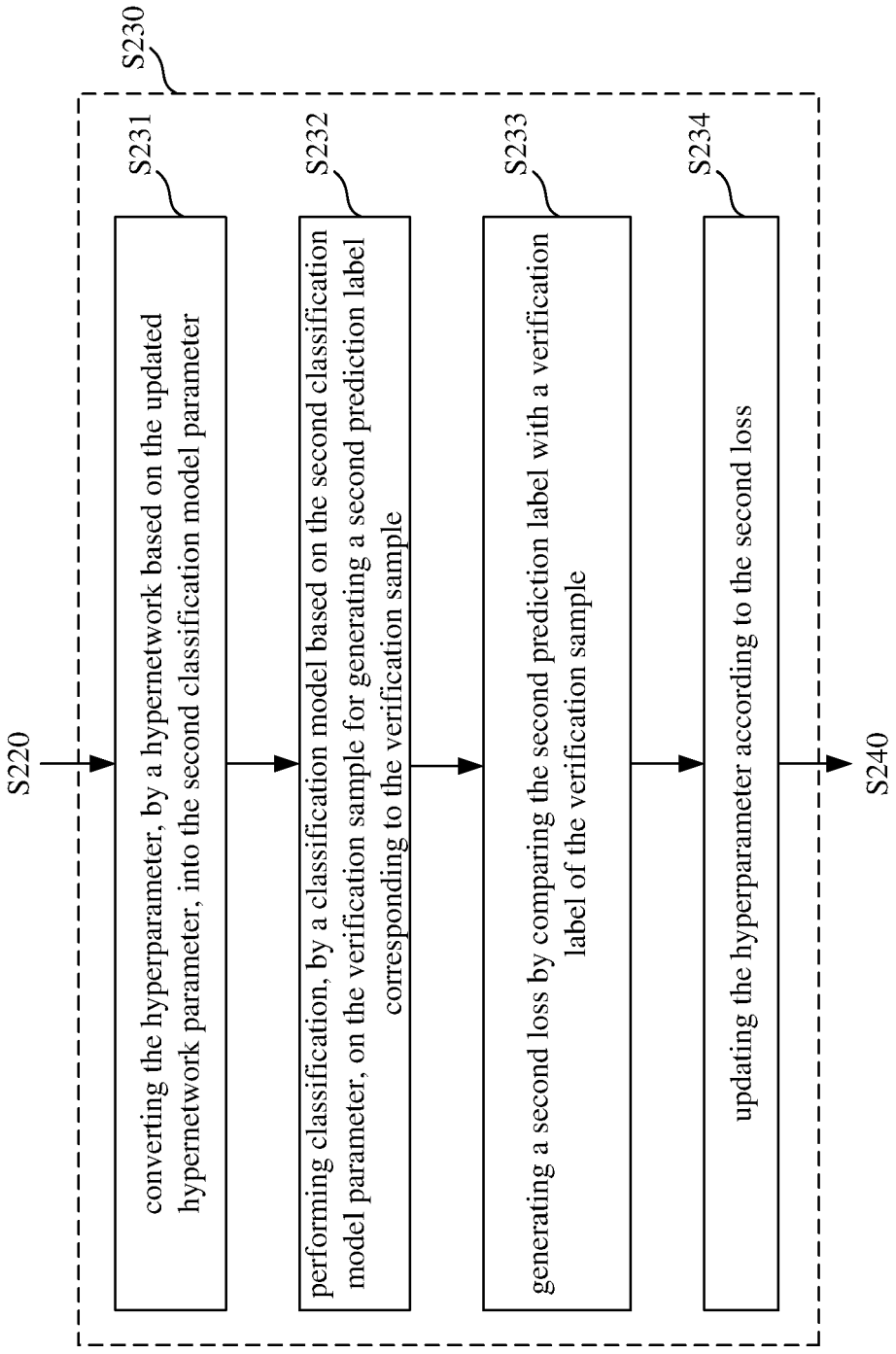

S220
S230
S231
converting the hyperparameter, by a hypernetwork based on the updated hypernetwork parameter, into the second classification model parameter
S232
performing classification, by a classification model based on the second classification model parameter, on the verification sample for generating a second prediction label corresponding to the verification sample
S233
generating a second loss by comparing the second prediction label with a verification label of the verification sample
S234
updating the hyperparameter according to the second loss
S240

FIG. 7

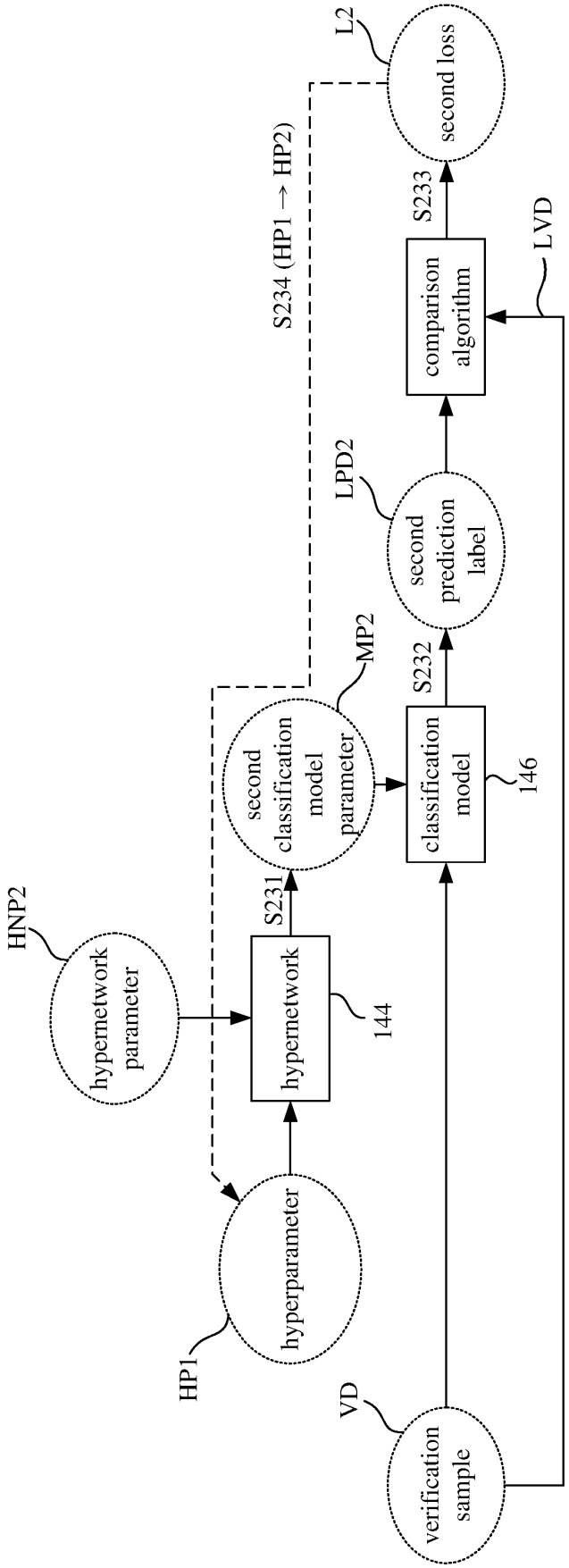

HNP2
hypernetwork parameter
hypernetwork
144
S231
second classification model parameter
MP2
S234 (HP1 → HP2)
L2
second loss
S233
comparison algorithm
LVD
LPD2
second prediction label
S232
classification model
146
HP1
hyperparameter
VD
verification sample

FIG. 8

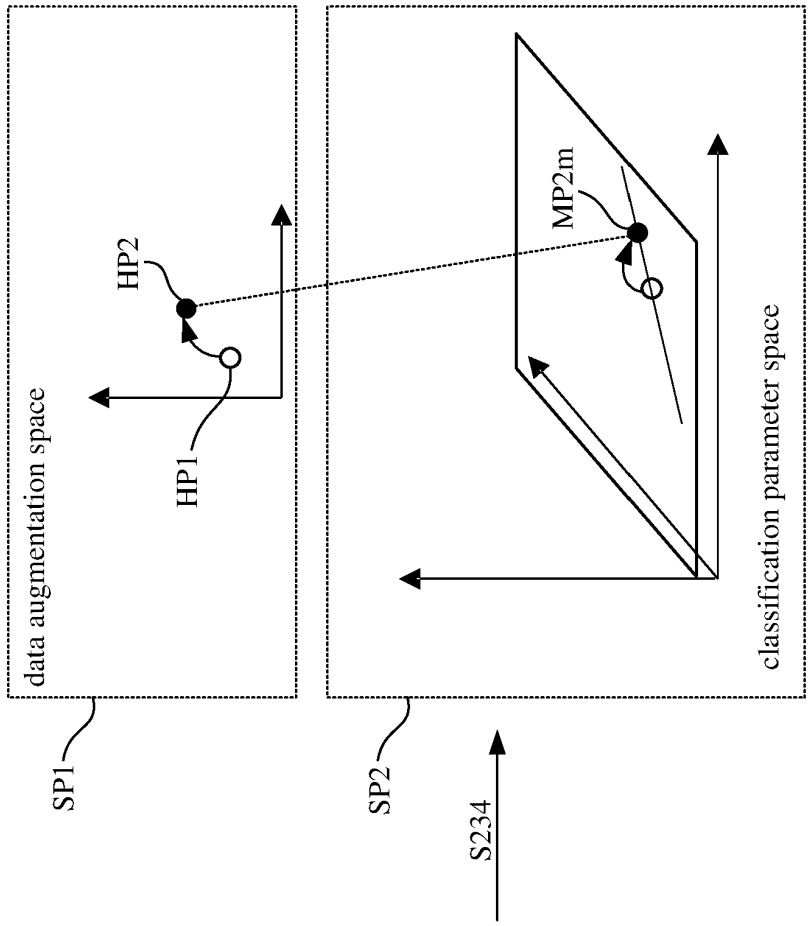

data augmentation space
HP2
HP1
SP1
SP2
S234
MP2m
classification parameter space

FIG. 9B

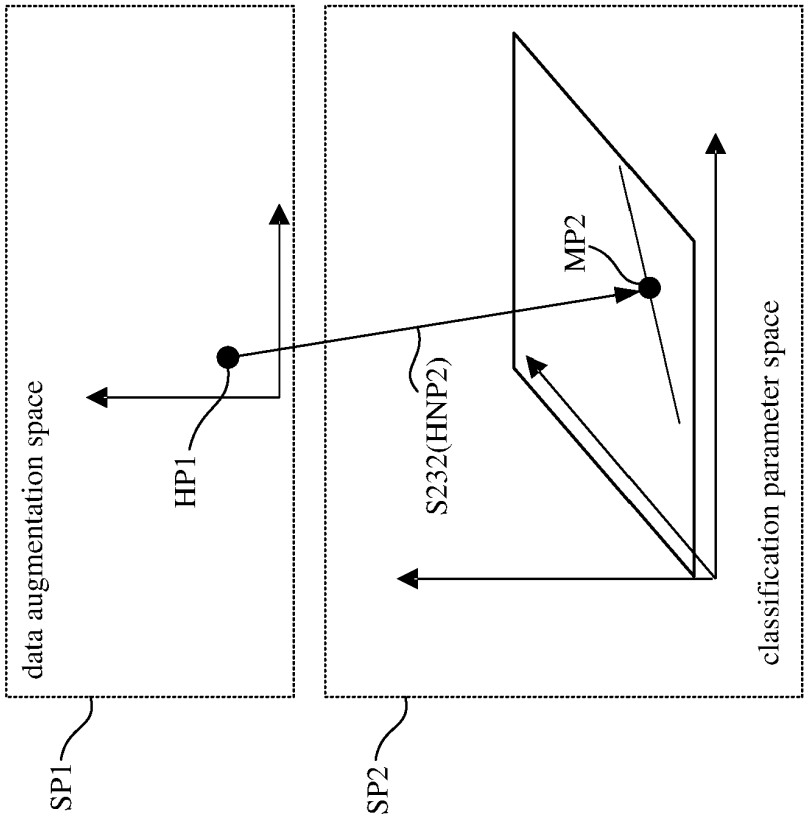

data augmentation space
HP1
SP1
S232(HNP2)
MP2
SP2
classification parameter space

FIG. 9A

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Non-patent literature cited in the description

- Generative Teachings Networks: Accelerating Neural Architecture Search by Learning to Generate Synthetic Training Date. **FELIPE PETROSKI SUCH.** Arxiv.org. Cornell University Library, 17 December 2019 **[0004]**
- **DAVID HA.** *HyperNetworks,* 01 December 2016, http://arxiv.org/pdf/1609.09106.pdf **[0004]**
- Self-Tuning Networks: Bilevel Optimization of Hyperparameters using structured best-response Functions. **MATTHEW MACKAY.** Arxiv.org. Cornell University Library, 07 March 2019 **[0004]**